



**Universidade do Estado do Rio de Janeiro**

Centro de Tecnologia e Ciências

Faculdade de Engenharia

Marcos Santana Farias

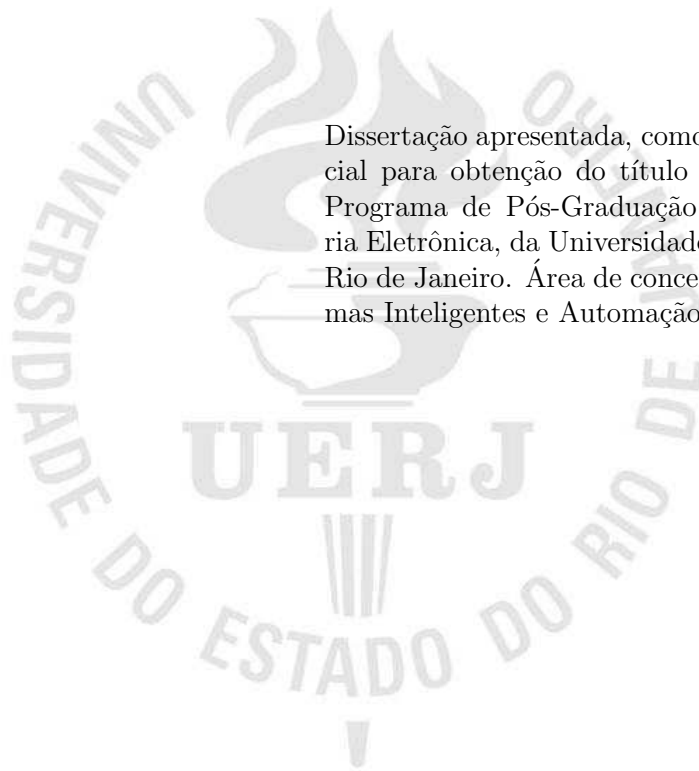
**Hardware reconfigurável para  
identificação de radionuclídeos utilizando  
o método de agrupamento subtrativo**

Rio de Janeiro  
2012

Marcos Santana Farias

**Hardware reconfigurável para  
identificação de radionuclídeos utilizando  
o método de agrupamento subtrativo**

Dissertação apresentada, como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Sistemas Inteligentes e Automação.



Orientadora: Prof<sup>a</sup>. Dr<sup>a</sup>. Nadia Nedjah

Co-orientadora: Prof<sup>a</sup>. Dr<sup>a</sup>. Luiza de Macedo Mourelle

Rio de Janeiro  
2012

CATALOGAÇÃO NA FONTE  
UERJ/REDE SIRIUS/CTC/B

F224 Farias, Marcos Santana

Hardware reconfigurável para identificação de radionuclídeos utilizando método de agrupamento subtrativo/Marcos Santana Farias. – 2012.

111f. : il.

Orientador: Nadia Nedjah.

Co-orientador: Luiza de Macedo Mourelle.

Dissertação (mestrado) – Universidade do Estado do Rio de Janeiro, Faculdade de Engenharia.

1. (Engenharia Eletrônica). 2. Hardware - Dissertações. 3. Radionuclídeos - Dissertações I. Nedjah, Nadia. II. Mourelle, Luiza de Macedo. III. Universidade do Estado do Rio de Janeiro. IV. Título.

CDU 004.3:539.163

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta dissertação, desde que citada a fonte.

---

Assinatura

---

Data

Marcos Santana Farias

# **Hardware reconfigurável para identificação de radionuclídeos utilizando o método de agrupamento subtrativo**

Dissertação apresentada, como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Sistemas Inteligentes e Automação.

Aprovado em 27 de Fevereiro de 2012

Banca Examinadora:

---

Prof<sup>a</sup>. Dr<sup>a</sup>. Nadia Nedjah (Orientadora)  
Faculdade de Engenharia, UERJ

---

Prof<sup>a</sup>. Dr<sup>a</sup>. Luiza de Macedo Mourelle (Co-orientadora)  
Faculdade de Engenharia, UERJ

---

Prof. Dr. Estevam Rafael Hruschka Junior  
Departamento de Computação, UFSCAR

---

Prof. Dr. Isaac José Antonio Luquetti dos Santos  
Instituto de Engenharia Nuclear, CNEN

Rio de Janeiro  
2012

# DEDICATÓRIA

Dedico esta dissertação à minha esposa, pelo apoio constante.

Aos meus pais, por onde tudo começou.

Aos irmãos e aos parentes, pelo incentivo.

Aos meus sogros, pelo apoio.

Aos meus amigos, pelas opiniões.

## **AGRADECIMENTOS**

Agradeço a DEUS por me abençoar e me capacitar.

Agradeço à minha esposa Sandra pelo apoio, indispensável e decisivo.

Agradeço à Professora Doutora Nadia Nedjah e à Professora Doutora Luiza M. Mourelle pelo estímulo e pelo conhecimento compartilhado.

Aos amigos pelo apoio, em especial aos amigos Mauro V. de Oliveira e José Carlos S. de Almeida pelo conhecimento compartilhado.

## RESUMO

Farias, Marcos Santana. *Hardware Reconfigurável para Identificação de Radionuclídeos Utilizando o Método de Agrupamento Subtrativo*. 2012. 111f. Dissertação (Mestrado em Engenharia Eletrônica) – Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2012.

Fontes radioativas possuem radionuclídeos. Um radionuclídeo é um átomo com um núcleo instável, ou seja, um núcleo caracterizado pelo excesso de energia que está disponível para ser emitida. Neste processo, o radionuclídeo sofre o decaimento radioativo e emite raios gama e partículas subatômicas, constituindo-se na *radiação ionizante*. Então, a *radioatividade* é a emissão espontânea de energia a partir de átomos instáveis. A identificação correta de radionuclídeos pode ser crucial para o planejamento de medidas de proteção, especialmente em situações de emergência, definindo o tipo de fonte de radiação e seu perigo radiológico. Esta dissertação apresenta a aplicação do método de agrupamento subtrativo, implementada em *hardware*, para um sistema de identificação de elementos radioativos com uma resposta rápida e eficiente. Quando implementados em software, os algoritmos de agrupamento consomem muito tempo de processamento. Assim, uma implementação dedicada para *hardware* reconfigurável é uma boa opção em sistemas embarcados, que requerem execução em tempo real, bem como baixo consumo de energia. A arquitetura proposta para o *hardware* de cálculo do agrupamento subtrativo é escalável, permitindo a inclusão de mais unidades de agrupamento subtrativo para operarem em paralelo. Isso proporciona maior flexibilidade para acelerar o processo de acordo com as restrições de tempo e de área. Os resultados mostram que o centro do agrupamento pode ser identificado com uma boa eficiência. A identificação desses pontos pode classificar os elementos radioativos presentes em uma amostra. Utilizando este *hardware* foi possível identificar mais do que um centro de agrupamento, o que permite reconhecer mais de um radionuclídeo em fontes radioativas. Estes resultados revelam que o *hardware* proposto pode ser usado para desenvolver um sistema portátil para identificação radionuclídeos.

**Palavras-chave:** Agrupamento subtrativo, *hardware* reconfigurável, identificação de radionuclídeos.

# ABSTRACT

Radioactive sources include radionuclides. A radionuclide is an atom with an unstable nucleus, i.e. a nucleus characterized by excess of energy, which is available to be imparted. In this process, the radionuclide undergoes radioactive decay and emits gamma rays and subatomic particles, constituting the ionizing radiation. So, radioactivity is the spontaneous emission of energy from unstable atoms. Correct radionuclide identification can be crucial to planning protective measures, especially in emergency situations, by defining the type of radiation source and its radiological hazard. This project introduces the application of subtractive clustering method, in a hardware implementation, for an identification system of radioactive elements that allows a rapid and efficient identification. In software implementations, clustering algorithms, usually, are demanding in terms of processing time. Thus, a custom implementation on reconfigurable hardware is a viable choice in embedded systems, so as to achieve real-time execution as well as low power consumption. The proposed architecture for the hardware of subtractive clustering is scalable, allowing for the inclusion of more of subtractive clustering unit that operate in parallel. This provides greater flexibility to accelerate the hardware with respect to the time and area requirements. The results show that the expected cluster center can be identified with efficiency. The identification of these points can classify the radioactive elements present in a sample. Using the designed hardware, it is possible to identify more than one cluster center, which would lead to the recognition of more than one radionuclide in radioactive sources. These results reveal that the proposed hardware to subtractive cluster can be used to design a portable system for radionuclides identification.

**Keywords:** Subtractive clustering, reconfigurable hardware, radionuclide identification.



## LISTA DE FIGURAS

1	Diferentes emissões de radiação pelo núcleo instável (HENRIKSEN; MAILLIE, 2003)	20
2	Emissão de radiação gama pelo núcleo . . . . .	21
3	Detector cintilador de NaI(Tl) . . . . .	22
4	Sistema de espectrometria gama . . . . .	22
5	Analizador multicanal . . . . .	23
6	Exemplo de agrupamento. . . . .	27
7	Exemplo de dendrograma de agrupamento hierárquico . . . . .	29
8	Espectro de energia simulado de uma fonte com Cs-137 e Co-60. . . . .	35
9	Diagrama da macro-arquitetura: UAS e UCAC . . . . .	38
10	Unidade de agrupamento subtrativo - UAS . . . . .	40
11	Exponencial natural $e^{-v}$ para $v \geq 0$ . . . . .	45
12	Polinômios quadráticos aproximadores a $e^{-v}$ , em $[0,8]$ . . . . .	46
13	Erro absoluto da aproximação por polinômios . . . . .	46
14	Unidade de Carga, Armazenamento e Controle - UCAC . . . . .	47
15	Máquina de estados do controlador principal . . . . .	50
16	Início da simulação . . . . .	51
17	Simulação no fim do cálculo nos componentes EXP. . . . .	52
18	Simulação no fim do cálculo nos componentes EXP nos endereços 5 e 6. . . . .	53
19	Simulação no fim do cálculo nos componentes EXP nos endereços 9 e 10. . . . .	54
20	Simulação no fim do somatório e gravação dos dados . . . . .	54
21	Simulação no momento do reconhecimento do fim dos dados na memória MD. . . . .	55
22	Gravação dos dados e início da redução de potencial . . . . .	56
23	Fim do primeiro cálculo exponencial na redução de potencial. . . . .	57
24	Fim da primeira redução de potencial e gravação do resultado . . . . .	58
25	Arquitetura da unidade Exp. . . . .	62
26	Unidade de <i>hardware</i> COEFICIENTES . . . . .	65
27	Máquina de estados do controlador do componente EXP. . . . .	67
28	Início da operação nos componentes EXP . . . . .	71
29	Simulação na busca dos coeficientes nos componentes EXP . . . . .	72
30	Fim do processamento nos componentes EXP . . . . .	72
31	Arquitetura interna do componente SOMADOR . . . . .	74
32	Máquina de estados do controlador do componente SOMADOR . . . . .	77
33	Início da operação no componente somador . . . . .	80
34	Final da primeira soma no componente somador . . . . .	81
35	Final do cálculo do primeiro potencial no componente somador . . . . .	81
36	Final do cálculo do último potencial no componente somador . . . . .	82
37	Início da redução de potencial no componente somador . . . . .	83
38	Escalabilidade da arquitetura. . . . .	84

---

39	Ilustração da escalabilidade da arquitetura. . . . .	85
40	Início da simulação . . . . .	87
41	Fim do cálculo exponencial nas UAS. . . . .	87
42	Fim do somatório e gravação de resultados na memória . . . . .	88
43	Início do cálculo exponencial na simulação. . . . .	91
44	Final do cálculo exponencial . . . . .	92
45	Final da primeira soma . . . . .	92
46	Final do primeiro somatório e do cálculo do primeiro potencial . . . . .	93
47	Final do segundo somatório . . . . .	93
48	Final do cálculo do último potencial . . . . .	94
49	Espectros testados . . . . .	95
50	Área ocupada em função do número de UAS na <i>Spartan 3E</i> . . . . .	99
51	Área ocupada em função do número de UAS <i>Vertex 6</i> . . . . .	100
52	Gráfico de testes realizados com 7 equipamentos . . . . .	102

## LISTA DE TABELAS

1	Radionuclídeos testados . . . . .	94
2	Comparação no número de ciclos para processamento . . . . .	97
3	Área ocupada e frequência máxima de operação em FPGA <i>Vertex</i> 6. . . . .	100
4	Comparação de resultados com equipamentos de identificação de radionuclídeos. . . . .	103

## LISTA DE ALGORITMOS

1	<i>K</i> -Médias . . . . .	30
2	Agrupamento Subtrativo . . . . .	34

# SUMÁRIO

INTRODUÇÃO .....	13
<b>1 IDENTIFICAÇÃO DE RADIONUCLÍDEOS .....</b>	<b>18</b>
1.1 Radionuclídeos e Tipos de Radiação .....	18
1.2 Detecção de Radiação Gama .....	21
1.3 Métodos e Utilidade .....	23
1.4 Considerações Finais do Capítulo .....	24
<b>2 MÉTODO DE AGRUPAMENTO SUBTRATIVO .....</b>	<b>26</b>
2.1 Métodos de Agrupamento .....	26
2.1.1 <u>Agrupamento Hierárquico</u> .....	28
2.1.2 <u>Agrupamento por Partição</u> .....	29
2.2 Agrupamento Subtrativo .....	31
2.3 Testes com Agrupamento Subtrativo .....	34
2.4 Considerações Finais do Capítulo .....	36
<b>3 MACRO-ARQUITETURA E MODELAGEM DOS DADOS .....</b>	<b>37</b>
3.1 Visão Geral .....	37
3.2 Componentes da Macro-Arquitetura .....	39
3.3 Aproximação por Polinômios e Representação numérica .....	42
3.3.1 <u>Representação numérica</u> .....	42
3.3.2 <u>Aproximação por polinômios</u> .....	44
3.4 Unidade de Carga, Armazenamento e Controle .....	45
3.5 Simulações .....	50
3.6 Considerações Finais do Capítulo .....	57
<b>4 ARQUITETURA DA UNIDADE DE AGRUPAMENTO .....</b>	<b>59</b>
4.1 Operações Aritméticas .....	59
4.2 Unidade para o Cálculo Exponencial .....	61
4.2.1 <u>Deslocamentos</u> .....	61
4.2.2 <u>Coeficientes para aproximação por polinômios</u> .....	63
4.2.3 <u>Controlador</u> .....	64
4.2.4 <u>Simulações</u> .....	70
4.3 Unidade para o Somatório .....	73
4.3.1 <u>Modelo proposto</u> .....	73
4.3.2 <u>Simulações</u> .....	80
4.4 Escalabilidade da Arquitetura .....	83
4.4.1 <u>Modelo escalável</u> .....	84
4.4.2 <u>Simulações</u> .....	86
4.5 Considerações Finais do Capítulo .....	88

---

<b>5</b>	<b>IMPLEMENTAÇÃO E RESULTADOS . . . . .</b>	<b>90</b>
<b>5.1</b>	<b>Resultados de Simulação . . . . .</b>	<b>90</b>
<b>5.2</b>	<b>Resultados de Processamento . . . . .</b>	<b>96</b>
<b>5.3</b>	<b>Implementação em FPGA . . . . .</b>	<b>98</b>
<b>5.4</b>	<b>Comparação com outras soluções . . . . .</b>	<b>101</b>
<b>5.5</b>	<b>Considerações Finais do Capítulo . . . . .</b>	<b>103</b>
<b>6</b>	<b>CONCLUSÃO E TRABALHOS FUTUROS . . . . .</b>	<b>104</b>
<b>6.1</b>	<b>Resumo e Conclusão . . . . .</b>	<b>104</b>
<b>6.2</b>	<b>Trabalhos Futuros . . . . .</b>	<b>105</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>107</b>

# INTRODUÇÃO

ESTA dissertação apresenta o desenvolvimento em *hardware* reconfigurável de um sistema automático para a identificação de radionuclídeos. Para esta tarefa foi proposto o uso de um método de agrupamento (*clustering*), denominado agrupamento subtrativo (CHIU, 1994).

Radionuclídeos são átomos com núcleos instáveis, ou seja, núcleos com excesso de energia que não pode ser mantida indefinidamente. A busca por um estado mais estável ocorre naturalmente com a emissão do excesso de energia em forma de partículas e ondas eletromagnéticas. Esse processo de alteração espontânea é chamado de *radioatividade*. As partículas e ondas eletromagnéticas emitidas neste processo recebem genericamente o nome de *radiação*. Ressalta-se entretanto que esta é uma *radiação ionizante*, ou seja, uma radiação que possui energia suficiente para ionizar átomos e moléculas. Essa característica torna a radiação ionizante potencialmente danosa aos seres vivos (GRUPEN, 2010).

O potencial de dano aos tecidos do ser humano, provocado pela radiação ionizante, foi descoberto nos primeiros anos após a descoberta dos raios X por Wilhelm Conrad Roentgen em 1895. Relatos de dermatites, perda de cabelo e anemia, associados ao uso desta nova tecnologia, começaram a aparecer, vitimando inclusive os primeiros pesquisadores da área (BELLINTANI; GILI, 2002). Estes fatos, junto com o aumento no uso da radiação ionizante na medicina e na indústria, levaram ao desenvolvimento dos estudos de *proteção radiológica* para se conhecer mais detalhes dos efeitos produzidos por doses repetidas de radiação a longo prazo.

O princípio básico da proteção radiológica estabelece que *todas as exposições a radiação devem ser mantidas tão baixas quanto razoavelmente exequíveis* (ALARA - *As Low As Reasonably Achievable*) (TAUHATA; SALATI, 2003). Isto significa que exposições a radiação ionizante devem ser controladas e justificadas, tanto para trabalhadores da área quanto para indivíduos do público. As normas para proteção radiológica são desenvolvidas internacionalmente por uma comissão, a ICRP (*International Commission on Radiological Protection*), e no Brasil pela CNEN (Comissão Nacional de Energia Nuclear).

Dentro da proteção radiológica, os estudos de identificação de radionuclídeos são importantes para a determinação do esquema de decaimento radioativo. No decaimento radioativo

um átomo “pai” se transforma em um átomo “filho” através da emissão de partículas e ondas eletromagnéticas, os raios  $\gamma$  (gama). Esse processo pode ocorrer várias vezes, transformando átomos em cadeia até que um “filho” seja estável. Equipamentos para determinação do esquema de decaimento são usados em laboratórios, com instrumentação específica para determinar o espectro de radiação das amostras. Esta instrumentação é chamada de *espectrômetros* de raios  $\gamma$  e inclui um detector de radiação, amplificador e conversor analógico digital.

A identificação de radionuclídeos também é importante para a proteção radiológica em atividades no campo. Monitores de radiação são usados para medir a dose de radiação presente em um ambiente, permitindo estabelecer os limites de tempo de exposição para trabalhadores ou identificando a presença de radiação ionizante. Em áreas de grande trânsito de pessoas e materiais, como portos e aeroportos, bem como em grandes eventos, a monitoração de radiação é aconselhável dentro das normas de proteção ao público (NOUAILHETAS, 2003), sendo obrigatória em instalações nucleares e radioativas (CNEN, 2011).

Monitores de radiação, no entanto, apenas detectam e quantificam a radiação presente em um ambiente. Embora esta seja a informação de maior importância para as atividades de proteção radiológica, muitas vezes a identificação rápida e automática do radionuclídeo presente é uma informação que ajuda na tomada de decisão. Por exemplo, em aeroportos a identificação correta de um radionuclídeo sendo transportado, mesmo que este esteja documentado, pode definir se o transporte é legal ou não.

Equipamentos para identificação de radionuclídeos no campo precisam ser portáteis. Existem hoje no mercado cerca de uma dezena de equipamentos que realizam esta tarefa. Mas a tarefa de identificação sofre a influência de muitos fatores, o que pode degradar a resposta desses equipamentos (BLACKADAR et al., 2006). Estes fatores afetam principalmente a calibração dos equipamentos, fazendo com que o espectro da energia detectada não seja corretamente interpretado.

Os métodos para identificação em geral se baseiam na busca no espectro pelos picos de energia  $\gamma$  característica que correspondem a cada radionuclídeo (KNOLL, 1989). A energia dos raios  $\gamma$ , que na maioria dos casos acompanha a emissão de partículas no processo de decaimento radioativo, é uma característica única dos radionuclídeos e é utilizada para detectá-lo e para identificá-lo.

O objetivo deste trabalho é contribuir para a melhoria de desempenho da tarefa de identificação de radionuclídeos, utilizando uma implementação em *hardware* do método de agrupamento subtrativo. O método de agrupamento, neste caso, é aplicado sobre o conjunto de



dados de energia detectada. Neste conjunto de dados, em caso de presença de um radionuclídeo na amostra sendo monitorada, os valores de energia característica deste elemento radioativo estarão presentes com mais frequência.

Os métodos de agrupamento procuram resolver o problema de criar grupos de dados, ou agrupamentos, de modo que os dados de um agrupamento possuam grande similaridade, enquanto dados de agrupamentos diferentes possuam baixa ou nenhuma similaridade (WISHART, 2002). A similaridade medida no caso deste trabalho é a diferença entre os valores de energia detectada. A frequência com que os valores de energia aparecem no conjunto de dados determina os agrupamentos. Logo, os valores de energia característica dos radionuclídeos definirá a formação de agrupamentos.

Os métodos de agrupamento de dados vão além da tarefa de classificação, em que objetos são distribuídos por diferentes grupos predefinidos. No agrupamento de dados, os grupos também têm de ser definidos (GAN; MA; WU, 2007). E esta definição dos grupos existentes no conjunto de dados é realizada de forma não supervisionada, o que torna os métodos de agrupamento inerentemente apropriados para aplicações *on-line* e em tempo real (FILEV; ANGELOV, 2007).

A escolha do método de agrupamento subtrativo para este trabalho foi feita, em primeiro lugar, por este ser um método onde não é necessário predefinir o número de agrupamentos que devem ser encontrados. Esta característica é útil porque não é possível definir quantos valores de energia característica estarão presentes em uma amostra radioativa. Outro fator de importância para a escolha do agrupamento subtrativo foi o fato deste ser de mais simples implementação em *hardware*.

O algoritmo para implementar o método de agrupamento subtrativo realiza um somatório da equação exponencial  $e^{-v}$ . Este somatório calcula o valor potencial de uma amostra do conjunto de dados. Este valor potencial é baseado na diferença do valor numérico desta amostra em relação a todas as outras amostras. Os valores numéricos das amostras são os valores de energia detectada. O algoritmo calcula o potencial de todas as amostras do conjunto de dados, escolhendo o de maior potencial como o centro do primeiro agrupamento encontrado.

A arquitetura de *hardware* proposta para este fim realiza a computação usando uma aproximação a polinômios quadráticos pelo Método dos Mínimos Quadrados (MMQ) (RUGGIERO; LOPES, 1988). Desta forma, um polinômio quadrático é ajustado à exponencial  $e^{-v}$  num intervalo definido de  $v$ . Foram escolhidas cinco faixas para aproximação:  $v \in [0, 1[$ ,  $v \in [1, 2[$ ,  $v \in [2, 4[$ ,  $v \in [4, 8[$  e  $v \in [8, +\infty[$ .

A arquitetura proposta buscou outra simplificação do *hardware* através da representação numérica de números reais em forma de frações de inteiros. Esta representação numérica é baseada no modelo *Fractional Fixed Point* (FFP) (SANTI-JONES; GU, 2008). Com este modelo é possível trabalhar com operações em números reais sem usar o formato de ponto flutuante (IEEE *Floating Point*) (TANENBAUM, 2007).

Com a representação numérica e a aproximação por polinômios escolhidas, o *hardware* foi desenvolvido para operar com operações aritméticas de soma, subtração e multiplicação de frações. Neste desenvolvimento buscou-se que as operações fossem executadas de forma paralela, sempre que possível, para a computação do valor polinômio. A unidade de *hardware* da arquitetura para realizar o somatório de valores exponenciais possui dois componentes para o cálculo exponencial e um para a soma e acumulação dos resultados provenientes dos dois primeiros. As máquinas de estados que os controla são independentes entre si, mas iniciadas por um controle principal.

Outras unidades para realizar o somatório de valores exponenciais podem ser implementadas em paralelo com primeira. Assim, vários somatórios podem ser realizados em paralelo. Essa escalabilidade da arquitetura proposta permite uma maior flexibilidade para a implementação do *hardware*, com uma possibilidade de diminuição no tempo de processamento do algoritmo.

Os componentes da arquitetura foram modelados em VHDL, uma linguagem especificamente usada para descrição de *hardware* e representação de sistemas digitais (NAVABI, 1997).

A principal contribuição deste trabalho foi o desenvolvimento da arquitetura de *hardware* básica de um equipamento portátil para a identificação de radionuclídeos, utilizando uma abordagem distinta das anteriores, baseada no agrupamento de dados. Contribuição importante também foi dada em demonstrar a aplicabilidade de soluções em *hardware* para a implementação de agrupamento de dados. Os resultados mostraram que os algoritmos de agrupamento, como o de agrupamento subtrativo, podem ser executados com grande melhora no desempenho em *hardware* dedicado, quando comparado a soluções em software.

Na organização desta dissertação, os dois primeiros capítulos são dedicados a introduzir os conceitos básicos de radioatividade e de agrupamentos. Os demais capítulos abordam a arquitetura proposta e discute os resultados.

O Capítulo 1 relata a importância da identificação de radionuclídeos, começando pela definição básica de radioatividade e como esta foi descoberta e passou a ter importância em atividades da medicina e da indústria. O Capítulo 1 explica também os tipos de radiação ioni-

zantes existentes e a detecção de radiação  $\gamma$ , a principal fonte de informação para a identificação de radionuclídeos.

O Capítulo 2 justifica a escolha pelo método de agrupamento subtrativo para a realização deste trabalho, introduzindo os conceitos básicos de agrupamento de dados e as características de alguns métodos. O Capítulo 2 também apresenta alguns testes com o método de agrupamento subtrativo para verificar se os resultados fornecidos são suficientes para que este possa ser usado em um sistema de identificação de radionuclídeos.

O Capítulo 3 fornece uma visão geral da arquitetura proposta, dando os detalhes da unidade de controle principal, que também inclui as memórias para armazenar as amostras e os resultados. Este capítulo também fornece uma explicação de como o *hardware* manipula os dados numéricos, usando a aproximação por polinômios e a representação numérica em forma de frações. Uma simulação da arquitetura, realizada no programa *ModelSim* (MENTOR-GRAPHICS, 2011), é apresentada, seguindo as etapas do controle principal.

O Capítulo 4 fornece as informações relacionadas as operações aritméticas que o *hardware* necessita realizar. Essas operações definem o número de multiplicadores e somadores para tornar a arquitetura a mais paralela possível. Os detalhes das unidades para o cálculo exponencial e para o somatório são apresentados, incluindo os detalhes da máquina de estados de cada unidade e as simulações para mostrar a atuação dos controles e do sincronismo de dados. Este capítulo também apresenta os detalhes para implementação da escalabilidade da arquitetura.

O Capítulo 5 apresenta os resultados obtidos com a simulação no *ModelSim*. Os resultados são apresentados e o processamento é avaliado quanto ao número de ciclos para a execução, comparando com uma solução em software embarcado. O capítulo mostra também os resultados de implementações em FPGA, com a análise da área ocupada em função do maior número de componentes de cálculo exponencial e somatório colocados em paralelo. Também é apresentada uma avaliação de identificadores de radionuclídeos presentes no mercado com uma breve comparação com resultados obtidos pelo *hardware* proposto.

O Capítulo 6 conclui a dissertação, apresentando a continuidade ao trabalho e outras aplicações onde o agrupamento de dados com implementação em *hardware* pode ser utilizado. Ressalta-se que esta dissertação é um prosseguimento de trabalhos anteriores que foram apresentados em dois congressos internacionais (FARIAS; NEDJAH; MOURELLE, 2011c, 2011b) e publicado em um periódico internacional (FARIAS; NEDJAH; MOURELLE, 2011a).

# Capítulo 1

## IDENTIFICAÇÃO DE RADIONUCLÍDEOS

A RADIAÇÃO constitui o transporte de energia através de um meio. Isso inclui todo o espectro eletromagnético de ondas, como as ondas de rádio, micro-ondas, luz, infravermelho visível, ultravioleta e raios X. Inclui também as partículas dotadas de massa, como as partículas alfa e beta. Nesta dissertação trataremos das *radiações ionizantes*, ou seja, que possuem energia suficiente para ionizar a matéria. Estas radiações podem arrancar elétrons de seus átomos, provocando a *ionização* e modificando as moléculas (TSOULFANIDIS, 1995). Exemplos desse tipo de radiação são os raios X, raios cósmicos, e as emissões a partir de elementos radioativos. Portanto, quando for mencionada somente a palavra “radiação” estaremos nos referindo a radiação ionizante.

Este capítulo apresenta os conceitos básicos da radioatividade e a importância da identificação dos elementos emissores dessa radiação nas atividades de proteção radiológica.

A Seção 1.1 apresenta como ocorreu a descoberta da radiação e os seus diferentes tipos, com mais atenção à radiação nuclear  $\gamma$  (gama), a mais utilizada para efetuar a detecção. A Seção 1.2 apresenta a instrumentação utilizada para avaliar espectros de radiação  $\gamma$  e onde se insere a identificação de radionuclídeos. A Seção 1.3 aborda a importância da identificação de radionuclídeos e as técnicas utilizadas. A Seção 1.4 apresenta um resumo do capítulo atual e introduz o assunto do Capítulo 2.

### 1.1 Radionuclídeos e Tipos de Radiação

Em novembro de 1895, o alemão Wilhelm Conrad Roentgen, estudando a fluorescência emitida por tubos de gás, descobriu os raios X. Com alguns dias de trabalho, Roentgen já havia observado as propriedades básicas dos raios X, seu poder de penetrar em materiais leves como papel e madeira, e sua absorção por materiais mais fortes, como alumínio e folhas de estanho

(TURNER, 2007). Ele descobriu que os raios afetam chapas fotográficas e fazem um eletroscópio carregado perder a sua carga, fenômeno esse causado pela ionização do ar provocada pelo raio X. Este foi o primeiro exemplo conhecido de radiação ionizante. No ano seguinte já havia aplicação da descoberta com a primeira unidade de radiografia diagnóstica nos Estados Unidos (BELLINTANI; GILI, 2002).

A emissão de radiação espontânea por elementos químicos, também chamada de radioatividade natural, foi descoberta em 1896 por Antoine Henri Becquerel quando trabalhava com sais de urânio. Pesquisas para isolar esses elementos se seguiram, com os primeiros resultados sendo observados por Marie Curie e Pierre Curie no isolamento dos elementos radioativos Polônio e Rádio. Desde o período dessas descobertas, novas técnicas que utilizam a radiação foram desenvolvidas nos diversos campos da atividade humana, possibilitando a execução de tarefas importantes e únicas na medicina, indústria e agricultura.

Como a radiação não pode ser vista nem sentida, é um desafio saber se ela está presente ou não em um ambiente. Quando não se tinha ainda o entendimento adequado sobre os efeitos biológicos das radiações ionizantes, muitos radiologistas morreram em consequência dos danos causados por estas (BELLINTANI; GILI, 2002). A detecção e medição das radiações são, portanto, fundamentais para a avaliação do grau de risco envolvido em atividades com exposição à radiação. Para saber como se detecta a radiação é preciso entender alguns conceitos e conhecer os tipos de radiação.

As radiações são emitidas por um átomo para que este chegue a um equilíbrio, pois tudo que existe na natureza tende a permanecer num estado estável. Os átomos instáveis, também chamados de radionuclídeos ou radioisótopos, necessitam passar por um processo de emissão do excesso de energia, a *radioatividade*. Portanto, a radioatividade é a alteração espontânea de um tipo de átomo em outro com a emissão de radiação para atingir a estabilidade (HENRIKSEN; MAILLIE, 2003). Quando a instabilidade está no núcleo atômico, ocorre o que é chamado de *radiação nuclear*.

Todos os elementos que contêm em seu núcleo o mesmo número de prótons, mas possuem números diferentes de nêutrons, têm as mesmas propriedades químicas (NOUAILHETAS, 2003). São chamados *isótopos*, elementos que têm o mesmo número atômico, mas número de massa diferentes. O número de isótopos de cada elemento é variável. Os isótopos de um mesmo elemento não tem as mesmas propriedades físicas. Assim, por exemplo, o isótopo do Iodo (I-127) é estável e todos os seus seis isótopos são radiativos, isto é, são radioisótopos (ANDREUCCI, 2010).

Os átomos instáveis podem emitir três tipos de radiação do núcleo, em um processo chamado *decaimento*:  $\alpha$  (alfa),  $\beta$  (beta) e  $\gamma$  (gama). Nessa emissão eles se transformam em outro elemento, que pode continuar sendo instável ou não. As radiações  $\alpha$  e  $\beta$  são partículas que possuem massa e são eletricamente carregadas.

A radiação  $\alpha$  é constituída de dois prótons e dois nêutrons, sendo portanto de carga positiva e baixa penetração em virtude da massa. Os raios  $\beta$  são elétrons provenientes do núcleo e podem ter carga negativa ou positiva (pósitrons). Os raios  $\gamma$ , assim como os raios X, são ondas eletromagnéticas com grande poder de penetração. Quando um núcleo decai por emissão de radiação  $\alpha$  ou  $\beta$ , em geral o núcleo residual está fora da configuração de equilíbrio, emitindo a energia excedente sob a forma de radiação  $\gamma$  (NOUAILHETAS, 2003). A Figura 1 mostra uma representação dessas emissões, ilustrando o seu poder de penetração com energias mais comuns.

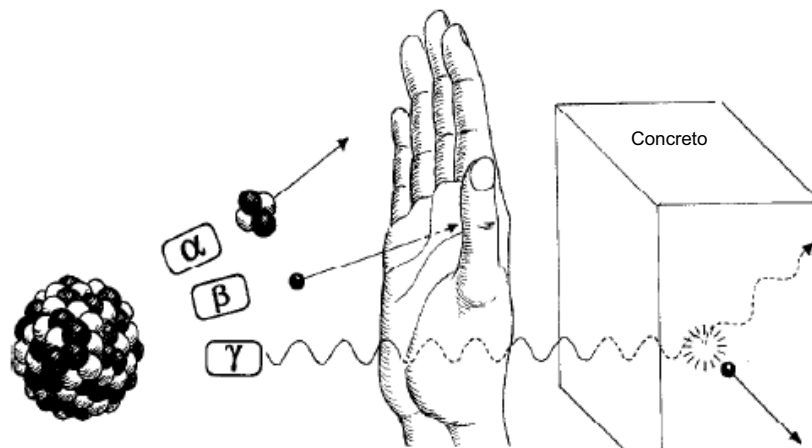


Figura 1: Diferentes emissões de radiação pelo núcleo instável (HENRIKSEN; MAILLIE, 2003)

A Figura 2 mostra a emissão da radiação  $\gamma$  pelo núcleo. Por ser bem definida, dependendo somente dos valores inicial e final de energia dos orbitais envolvidos na transição (TAUHATA; SALATI, 2003), a energia  $\gamma$  emitida ( $E_\gamma$ ) por um radionuclídeo pode caracterizá-lo e ser usada para identificá-lo, como será visto na Seção 1.2.

A energia da radiação é geralmente expressa em elétron-Volt (eV). Um eV é uma quantidade pequena de energia cinética adquirida por um elétron ao ser acelerado por uma diferença de potencial elétrica de 1 Volt (TAUHATA; SALATI, 2003).

O tempo que um núcleo radioativo leva para se transformar num estado mais estável é um dado probabilístico e dependente de suas características físicas. As transformações dos átomos instáveis, de mesma espécie, não são realizadas ao mesmo tempo, ocorrendo de modo

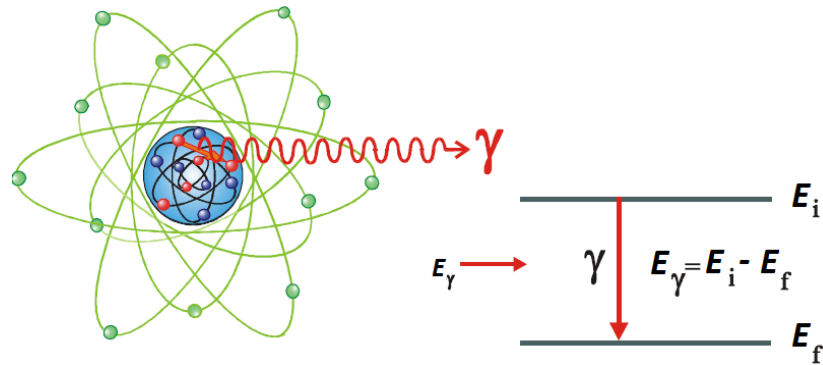


Figura 2: Emissão de radiação gama pelo núcleo

aleatório e sem haver possibilidade de se saber quando um determinado núcleo irá se transformar por decaimento. No entanto, considerando um número grande de átomos em uma amostra, a probabilidade de decaimento por átomo por segundo é uma constante, chamada constante de decaimento ( $\lambda$ ).

Essa constante irá definir a meia-vida  $T_{1/2}$  do radionuclídeo, ou seja, o tempo necessário para que a metade dos átomos radioativos da amostra decaiam, dado pelo Equação 1. A meia-vida é uma característica de um radionuclídeo, que pode variar de segundos a bilhões de anos.

$$T_{1/2} = 0,693\lambda \quad (1)$$

## 1.2 Detecção de Radiação Gama

A detecção de radiação gama é a principal fonte de informação sobre os radionuclídeos presentes em amostras radioativas. Desde que as descobertas sobre a radiação nuclear impulsionaram o desenvolvimento de várias atividades, se tornou imperativo realizar medições de radiação nuclear para a monitoração da quantidade de radiação presente no ambiente. Isto fez com que fosse criado um novo ramo da ciência, a *proteção radiológica*, com a finalidade de proteger os indivíduos, regulamentando e limitando o uso das radiações em condições aceitáveis (BELLINTANI; GILI, 2002).

Um sistema de detecção de radiação nuclear gama é constituído de duas partes: um dispositivo detector e outro de medida. A interação da radiação com o sistema ocorre no detector e o sistema de medida interpreta esta interação. A fim de detectar a radiação  $\gamma$ , detectores especiais são necessários, chamados de *cintiladores*. Esses detectores são capazes de emitir luz (fótons) quando a radiação ionizante transfere para eles toda ou parte da sua

energia. Essa luz é detectada por um fotomultiplicador, acoplado opticamente ao cintilador, que fornece à saída um sinal elétrico cuja amplitude é proporcional à energia depositada. Para a radiação gama, o cintilador mais utilizado é o cristal de Iodeto de Sódio ativado com Tório, NaI(Tl). A Figura 3 apresenta o conjunto detector.

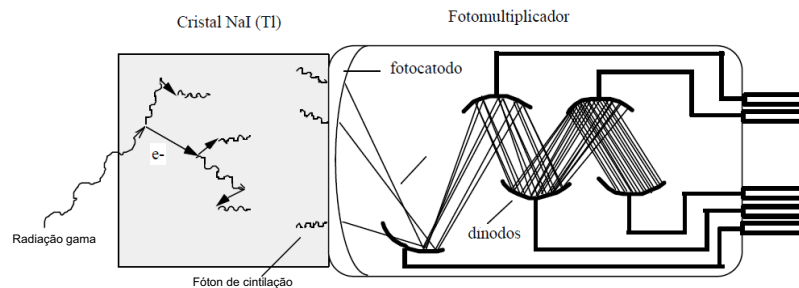


Figura 3: Detector cintilador de NaI(Tl)

A propriedade desses detectores de fornecer um sinal elétrico proporcional à energia depositada permite a criação de um espectro da energia gama emitida por um radionuclídeo. O equipamento utilizado em espectrometria gama inclui, além do detector de radiação, um amplificador e um classificador de altura de pulso, ou seja, um analisador multicanal. Este conjunto é mostrado na Figura 4 e tem por resultado um histograma armazenado na memória.

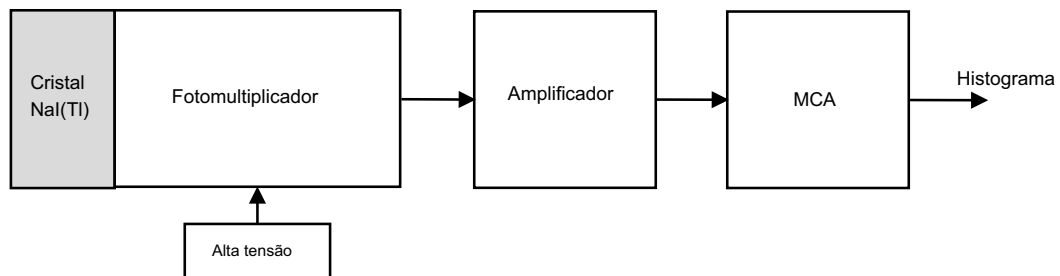


Figura 4: Sistema de espectrometria gama

A operação do analisador multicanal (*Multichannel Analyzer* - MCA) está baseada no princípio de converter um sinal analógico, a amplitude de um pulso, para um número digital equivalente. Uma vez que esta conversão é realizada, a tecnologia disponível para o armazenamento e exibição de informação digital é usada para registrar espectros de altura de pulso. Como resultado, o conversor analógico digital (*Analog to Digital Converter* - ADC) é um elemento fundamental, determinando as características de desempenho do analisador. A saída do ADC é armazenada em uma memória que tem tantos locais de endereçamento quanto o número máximo de canais nos quais o espectro registrado pode ser subdividido.



A função básica de um MCA envolve só o ADC e a memória. Com a finalidade de ilustração, pode-se imaginar a memória sendo somente organizada como uma pilha vertical de endereços de locação (Figura 5), variando do primeiro endereço (ou canal número 0) ao número de locação máxima (1023, neste caso). Uma vez que um pulso é processado pelo ADC, os circuitos de controle do analisador procuram o local de memória que corresponde à amplitude digitalizada, e o conteúdo daquele local é incrementado por um.

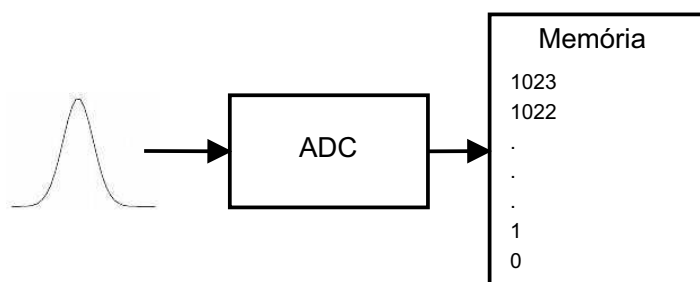


Figura 5: Analisador multicanal

Ao final a operação forma na memória um espectro de energia da amostra radioativa sob estudo. A visualização dos dados desta memória pode ser usada para os estudos de decaimento radioativo e como método para a identificação do radionuclídeo. Na Seção 1.3 será mostrada a importância da identificação de radionuclídeos e como é feita esta identificação na maioria das aplicações.

## 1.3 Métodos e Utilidade

A identificação correta do radionuclídeo é crucial para planejar medidas de proteção radiológica, especialmente em situações de emergência, pois permite definir o tipo da fonte de radiação e seu potencial de dano, despertando interesse também para aplicações em campo, em áreas de segurança, locais de grande circulação de pessoas e em grandes eventos. Isto em função de permitir uma tomada de decisão mais ágil, sem a necessidade de esperar por resultados que só poderiam ser conseguidos em laboratório.

Os esforços para uma correta identificação se concentram no desenvolvimento de detectores com maior resolução, no uso de um ADC estável e rápido e em bons amplificadores, com baixo ruído e boa conformação do sinal. Também se concentram na interpretação rápida e correta dos dados, o que no caso de sistemas portáteis é ainda mais importante. Neste trabalho buscamos contribuir para este último aspecto.

O espectro de energia formado no analisador multicanal de um sistema de espectrometria é a principal fonte de informação para os estudos sobre a estrutura dos radionuclídeos (CHUNG, 2001), sendo que uma análise detalhada desse espectro é normalmente usada para determinar a identidade e quantidade de emissores gama presentes na fonte radioativa. Cada radionuclídeo, dentro dos mais de mil conhecidos, possui um esquema de decaimento único pelo qual pode ser identificado (KAHN, 2006).

Na prática, a identificação do radionuclídeo é realizada por uma busca no espectro pelos valores listados de energias características que correspondem a cada radionuclídeo. Essa busca pode ser feita analisando visualmente o espectro e comparando as energias encontradas com as listadas em estudos anteriores. Também pode ser feita de forma automática por *software*.

Os métodos para encontrar as energias características, e identificar o radionuclídeo, sempre buscam os endereços da memória com o maior número de ocorrências, o que corresponde aos picos de energia do espectro. As variações nos algoritmos que realizam essa busca não são muito grandes e a maioria dos *softwares* podem chegar a bons resultados. Porém, quando se trata de aplicações no campo, com equipamentos portáteis de identificação de radionuclídeos, a resposta desses algoritmos, rodando em processadores de propósito geral, leva a um período de processamento muito grande para a aplicação. Em geral se espera que equipamentos portáteis, que operam em tempo real, respondam em intervalo de tempo não maior do que um segundo.

O que ocorre em alguns casos é que o algoritmo tem de ser simplificado, o que leva a erros de interpretação dos dados, com falsos positivos, falsos negativos ou desempenho comprometido com a identificação de apenas um radionuclídeo (BLACKADAR et al., 2006).

Neste trabalho buscamos um método de interpretação dos dados de energia, provenientes do sistema de detecção, que pudesse ser rápido e com bom desempenho, mesmo para aplicações portáteis. Foi desenvolvido para isso uma arquitetura de *hardware* que realize a classificação dos dados convertidos pelo ADC. Neste caso a memória não é incrementada em contagens nos endereços a cada ocorrência de uma determinada energia. A memória guarda os dados convertidos pelo ADC, em endereços subsequentes, durante um intervalo de tempo. A arquitetura proposta realiza a classificação através de um método de agrupamento, que será explicado no Capítulo 2.

## 1.4 Considerações Finais do Capítulo

Este capítulo estabeleceu os conceitos de radiação, radionuclídeos e a importância da identificação destes para as aplicações na área nuclear. Foi mostrado que os sistemas para identificação

encontram vários obstáculos para obter a correção nos resultados e que este trabalho visa contribuir para o melhor desempenho na interpretação dos dados dos detectores de radiação.

O próximo capítulo irá apresentar o algoritmo de agrupamento usado neste trabalho, verificando as alternativas estudadas e a justificativa para a escolha do algoritmo de agrupamento subtrativo.

## Capítulo 2

# MÉTODO DE AGRUPAMENTO SUBTRATIVO

**O**S algoritmos de agrupamento de dados (*clustering*) tornaram-se amplamente utilizados e bem aceitos para aplicações que precisam de classificação de dados. Áreas como processamento digital de imagens, reconhecimento de padrões e mineração de dados (*data mining*) são alguns exemplos de aplicações que exploram os conceitos e os algoritmos de agrupamento, tratados como ferramentas essenciais para o desenvolvimento de soluções específicas ou como facilitador para a interpretação dos dados.

Este capítulo tem a finalidade de apresentar os conceitos básicos de agrupamento, examinando alguns métodos utilizados em classificação de dados. Também tem a finalidade de justificar a escolha do método de agrupamento subtrativo para este trabalho, utilizando-o para classificar amostras de energia gama emitida por radionuclídeos.

A Seção 2.1 apresenta os conceitos básicos de agrupamento e alguns métodos/algoritmos de agrupamento mais utilizados. Na Seção 2.2 será apresentado o algoritmo de agrupamento subtrativo e as suas vantagens para a aplicação neste trabalho. A Seção 2.3 mostra os resultados de avaliação do método de agrupamento subtrativo. As considerações finais do capítulo estão na Seção 2.4.

### 2.1 Métodos de Agrupamento

O agrupamento de dados é uma tarefa, de aprendizado não-supervisionado, que objetiva a separação de um conjunto de dados em um certo número de grupos ou subconjuntos (OLIVEIRA; PEDRYCZ, 2007). Sendo uma tarefa não-supervisionada, os processos de classificação se desenvolvem sem a utilização de rótulos específicos para os objetos (JAIN; DUBES, 1988). Esta é uma razão que torna os métodos de agrupamento inerentemente apropriados para aplicações *on-line* (FILEV; ANGELOV, 2007). Métodos de agrupamento são descritos como aqueles usados

para dividir um conjunto de  $n$  amostras em  $k$  grupos, de tal forma que a similaridade entre membros do mesmo grupo é maior do que entre membros de grupos diferentes (RIPLEY, 1996).

Uma noção mais simples de agrupamento o define como um grupo com baixas distâncias entre os seus membros, o que significa áreas densas do espaço de dados. Portanto, métodos de agrupamento são dotados, em geral, de funções de distância para medir a dissimilaridade das amostras, o que é equivalente a medir a sua semelhança (KRUSE; DORING; LESOT, 2007). Um exemplo gráfico pode ser visto na Figura 6, onde é possível identificar três agrupamentos, cujo o critério de similaridade é a distância entre eles.

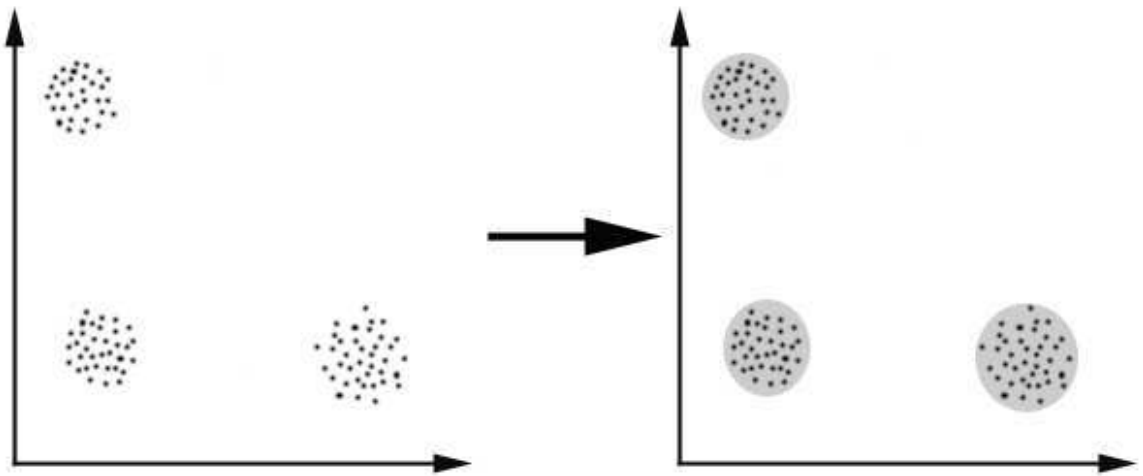


Figura 6: Exemplo de agrupamento

Alguns autores fazem distinção entre conjunto de dados *objetos* e conjunto de dados *relacionais*, considerando que dados *objetos* (*object data*) contêm um vetor numérico das características de cada objeto, como comprimento, peso, energia, etc. Dados *relacionais* quantificam a relação entre cada par de objetos, como por exemplo, a similaridade de dois objetos (RUNKLER, 2007). Porém, conjuntos de dados *relacionais* podem ser representados por dados *objetos*. Por exemplo, os objetos podem ser colocados em um diagrama bi-dimensional, de modo que pares mais similares são colocados mais próximos.

Neste trabalho o conjunto de dados utilizados possui uma dimensão, os valores numéricos que representam a energia da radiação gama detectada, como explicado no Capítulo 1. O critério de similaridade é, portanto, a diferença entre as energias.

Há muitas maneiras para expressar e formular um problema de agrupamento, e como consequência, os resultados obtidos e suas interpretações dependem fortemente da forma como o problema de agrupamento foi originalmente formulado. As diferentes formas de formular um

problema levaram a criação de uma vasta coleção de algoritmos para a tarefa de agrupamento, o que pode confundir um usuário a tentar selecionar um algoritmo mais apropriado para o seu problema (JAIN; MURTY; FLYNN, 1999).

Embora existam na literatura muitas taxonomias para métodos de agrupamento, a classificação mais tradicional divide-os em duas classes principais, com base nas propriedades dos agrupamentos gerados (EVERITT; LANDAU; LEESE, 2001):

1. Agrupamento Hierárquico (*Hierarchical Clustering*);
2. Agrupamento por Partição (*Partitional Clustering*).

Alguns algoritmos de agrupamento, pertencentes a essas classes, foram considerados para a escolha do algoritmo utilizado neste trabalho. As seções seguintes apresentam uma visão geral destas classes e de alguns métodos, para que melhor se compreenda a escolha do método de agrupamento subtrativo.

### 2.1.1 Agrupamento Hierárquico

O agrupamento hierárquico constrói uma hierarquia de grupos ou, em outras palavras, uma árvore de grupos, também conhecida como *dendrograma*. Cada nó de grupo contém aglomerados “filhos”, grupos de aglomerados irmãos dos pontos abrangidos pelo seu “pai” comum (ZHAO; KARYPIS, 2002). Esse método é dividido em aglomerativo (*agglomerative*) e divisivo (*divisive*) (KAUFMAN; ROUSSEEUW, 1990), como mostrado na Figura 7, onde os valores indicam diferentes amostras do conjunto de dados. Verifica-se que a direção de agrupamento para o método divisivo é oposto daquele do método aglomerativo. O agrupamento aglomerativo começa com um grupo de um ponto e recursivamente associa mais grupos. O agrupamento divisivo começa com um grupo com todos os pontos e segue separando os grupos apropriadamente.

Esta representação por dendrogramas fornece descrições informativas e uma visualização das estruturas de agrupamento de dados em potencial, especialmente quando existem reais relações hierárquicas entre os dados, como por exemplo em pesquisas evolucionárias em diferentes espécies de organismos (THEODORIDIS; KOUTROUMBAS, 2006). O dendrograma da Figura 7 pode ser subdividido em um nível segundo um critério de parada, representado pela linha pontilhada, para obter diferentes agrupamentos de dados.

Embora possua vantagens, como a flexibilidade em relação ao nível de granularidade e a facilidade de manuseio de qualquer forma de semelhança ou distância, existem críticas a

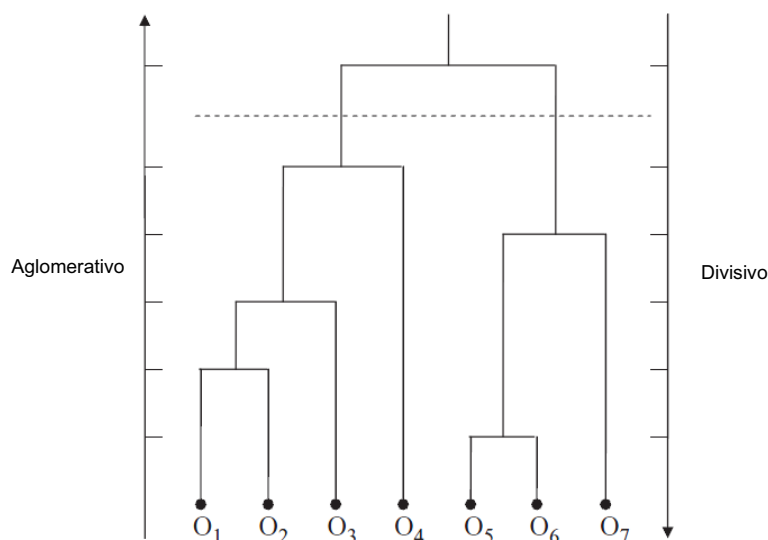


Figura 7: Exemplo de dendrograma de agrupamento hierárquico

estes métodos de agrupamento hierárquico em relação a alta complexidade computacional (XU; WUNSCH, 2009). Isto limita a sua aplicação em conjuntos de dados grandes.

Métodos de agrupamento hierárquico possuem ampla aplicação em áreas onde a classificação dos dados é inerentemente hierárquica (SHARAN; SHAMIR, 2000) como em dados de expressão genética (EISEN et al., 1998; YEUNG; MEDVEDOVIC; BUMGARNER, 1998), classificação de documentos (WILLETT, 1988) e agrupamento de textos na WEB (CHEN; ALAHAKOON; INDRAWAN, 2005).

### 2.1.2 Agrupamento por Partição

Os métodos de agrupamento por partição procuram diretamente uma partição dos  $n$  objetos, ao invés de uma estrutura de agrupamento, tais como o dendrograma produzido pela técnica hierárquica. Estes métodos têm vantagens em aplicações que envolvam grandes conjuntos de dados para os quais a construção de um dendrograma é computacionalmente proibitivo. No entanto um problema que acompanha o uso destes algoritmos é a necessidade da escolha antecipada do número desejado de partições de saída (XU; WUNSCH, 2009).

Um dos mais conhecidos métodos desta classe é o  $K$ -Médias ( $K$ -Means) (MACQUEEN, 1967). Assim como os algoritmos de agrupamento hierárquicos, este é um método de agrupamento do tipo *exclusivo*, ou seja, os dados são agrupados de modo que uma amostra pertença a um grupo exclusivamente. O procedimento para o algoritmo de agrupamento  $K$ -Médias segue uma maneira simplificada de classificar um determinado conjunto de dados através de um certo

**Algoritmo 1** *K*-Médias

- 
- 1: **Entrada**  $k$  ( $w_1, w_2, \dots, w_k$ ) com  $w_j$  conforme Equação 2.
  - 2: **Saída** Erro  $E$ .
  - 3: Cada grupo  $C_j$  é associado com  $w_j$
  - 4: **Repete**
  - 5: **Para**  $i_l := 1 : n$  **Faça**
  - 6:   Transfira  $i_l$  para o agrupamento  $C_{j^*}$  com o  $w_{j^*}$  mais próximo,
  - 7:   ou seja,  $|i_l - w_{j^*}| \leq |i_l - w_j|$ , com  $j \in \{1, 2, \dots, k\}$ ;
  - 8: **Fim Para**
  - 9: **Para**  $C_j := 1 : k$  **Faça**
  - 10:   Atualize o  $w_j$  para definir o centroide de todas as amostras em  $C_j$ ,
  - 11:   de modo que  $w_j$  seja calculado usando a Equação 4;
  - 12: **Fim Para**
  - 13: Calcule a função de erro usando a Equação 3;
  - 14: **Até**  $E$  não mudar mais significativamente.
- 

número de  $k$  grupos fixados *a priori*.

Com os  $k$  grupos ( $w_1, w_2, \dots, w_k$ ) sendo iniciados para uma das  $n$  amostras ( $i_1, i_2, \dots, i_n$ ), temos:

$$w_j = i_l, j \in \{1, 2, \dots, k\}, l \in \{1, 2, \dots, n\} \quad (2)$$

O Algoritmo 1 mostra como o método é processado. O objetivo do algoritmo é minimizar a função objetivo de erro quadrático dada pela Equação 3.

$$E = \sum_{j=1}^k \sum_{i_l \in C_j} |i_l - w_j|^2, \quad (3)$$

onde  $C_j$  é o  $j$ ésimo agrupamento cujo valor é um subconjunto disjunto das amostras de entrada e pode ser calculado usando a Equação 4.

$$C_j = \frac{1}{w_j} \sum_{i_l \in C_j} i_l, \quad (4)$$

A ideia de definir uma função objetivo e ter a sua minimização conduzindo o processo de agrupamento é bastante universal. Muitas extensões e modificações tem sido propostas visando melhorias dos resultados de agrupamento no que diz respeito a problemas específicos, como por exemplo, o ruído. Consequentemente, outras funções objetivo foram adaptadas para estas aplicações específicas (OLIVEIRA; PEDRYCZ, 2007).

Extensões dos métodos de agrupamento por partição também levaram ao desenvolvimento de algoritmos que diferem quanto ao modo de atribuir amostras como pertencentes a um grupo. Os métodos por partição *exclusivos*, ou *rígidos*, vistos até aqui, forçam a atribuição



de cada amostra para um, e somente um, dos grupos. E isto pode ser inadequado para amostras que são quase equidistantes de dois ou mais grupos. Por este motivo algumas variações dos métodos de agrupamento por partição foram desenvolvidas com uma abordagem nebulosa (*fuzzy*), onde amostras podem pertencer a mais de um grupo com diferentes graus de pertinência (ZADEH, 1965). Este tipo de atribuição pode representar a estrutura do aglomerado de forma mais natural, especialmente quando estes se sobrepõem (KRUSE; DORING; LESOT, 2007). O algoritmo de agrupamento FCM (*Fuzzy C-Means*) (HATHAWAY; HU, 1993) é uma extensão do algoritmo  $K$ -Médias que utiliza essa abordagem nebulosa.

O algoritmo de agrupamento subtrativo, que será apresentado na Seção 2.2, também foi criado com uma abordagem nebulosa, para ser usado na extração de regras de classificação a partir de dados numéricos (CHIU, 1994).

## 2.2 Agrupamento Subtrativo

Na Seção 2.1, a descrição sucinta de alguns métodos de agrupamento revelou algumas características não desejadas para este trabalho. A primeira delas diz respeito aos métodos de agrupamento por partição (Seção 2.1.2) que exigem, como parâmetro para o algoritmo ser executado, o número de agrupamentos que deve ser encontrado.

Foi visto no Capítulo 1 que a tarefa de identificar radionuclídeos passa pela classificação dos valores de energia gama depositada no detector. A presença de um radionuclídeo próximo a este detector ocasiona o aparecimento de um ou mais valores de energia, característica desse radionuclídeo, com maior frequência neste conjunto de dados. A quantidade desses valores de energia característica varia para diferentes radionuclídeos. Também o número de radionuclídeos em uma amostra radioativa pode variar. Portanto, a quantidade de valores de energia característica em um conjunto de dados, que representa o número de agrupamentos, pode variar bastante e não deveria ser predefinido. Caso fossem, agrupamentos que identificam um radionuclídeo poderiam ser não reconhecidos, e outros reconhecidos poderiam identificar elementos não presentes na amostra radioativa.

Situações como essa podem ser contornadas, e mesmo o uso destes algoritmos poderiam trazer bons resultados na identificação de radionuclídeos, mas certamente ao custo de mais processamento e maior complexidade, exigindo grande esforço computacional nas sucessivas tentativas para encontrar o número adequado de agrupamentos. Por essa razão foi feita a escolha por um algoritmo que não necessitasse desse parâmetro.

A segunda característica diz respeito aos métodos de agrupamento hierárquico. Os

algoritmos de agrupamento em geral exigem muitos cálculos iterativos, estendendo o tempo de processamento para chegar a um resultado. Entretanto alguns, desconsiderando a eficiência na obtenção do resultado, são excessivamente complexos. Um exemplo é o método divisivo do agrupamento hierárquico. Este necessita considerar até  $2^{N-1} - 1$  possíveis divisões em um agrupamento de  $N$  amostras (XU; WUNSCH, 2009). O método aglomerativo, menos custoso computacionalmente do que o divisivo, também apresenta intenso aumento de iterações com o aumento no número de amostras.

A escolha do método de agrupamento a ser utilizado baseou-se, portanto, na tentativa de encontrar características de maior simplicidade computacional, sem a necessidade de parâmetros de entrada que limitassem *a priori* o número de agrupamentos encontrados. O algoritmo de agrupamento subtrativo foi o que mais se aproximou dessas características.

O método de agrupamento de dados subtrativo é uma forma modificada do método da montanha (YAGER; FILEV, 1993). No método da montanha, supondo  $n$  pontos  $\{x_1, \dots, x_n\}$ , são definidas grades, igualmente espaçadas, para o espaço de dados, com  $v_i$  sendo  $i$ -ésimo vértice dessas grades. No método da montanha os centros de agrupamento são restritos aos vértices  $v_i$  da grade, ou seja, os centros iniciais são formados pelos vértices da grade. A função montanha para cada vértice  $v_i$  é dada pela Equação 5.

$$M(v_i) = \sum_{j=1}^n e^{-\alpha \|x_j - x_i\|^2}, \quad (5)$$

onde  $\alpha > 0$ ,  $M$  é a função montanha calculada para o  $i$ -ésimo vértice  $v_i$  durante o primeiro passo e  $n$  é número total de dados. O módulo  $\|x_j - x_i\|$  denota a distância euclidiana entre os pontos usados e  $x_i$ , o ponto ou amostra atual. É assegurado que o vértice em torno de muitos pontos terá um alto valor para esta função e, inversamente, um vértice com poucos pontos na vizinhança terá um baixo valor para a mesma função.

A função montanha é usada somente durante o primeiro passo com todo o conjunto de dados disponíveis. Durante os passos seguintes, a função é definida pela subtração de um valor proporcional ao valor de pico da função montanha.

O método de agrupamento subtrativo (CHIU, 1994) usa uma função similar ao método da montanha, onde o valor da função é chamado de valor potencial, definido pela Equação 6.

$$P_i = \sum_{j=1}^n e^{-\alpha \|x_j - x_i\|^2}, \text{ onde } \alpha = \frac{4}{r_a^2} \quad (6)$$

Na Equação 6,  $P_i$  é o valor potencial no ponto  $i$ ,  $x_i$  é o ponto atual e  $r_a$  uma constante positiva, chamada de *raio do agrupamento*. O valor  $r_a$  tem influência no número de centros

de agrupamentos encontrados. Com um raio  $r_a$  menor, o número de agrupamentos tende a ser maior. Se o raio  $r_a$  for grande o suficiente para incluir todo o conjunto de dados, somente um agrupamento será encontrado. Para este projeto o valor de  $r_a$  deve ser definido para que mais de um centro de agrupamento seja encontrado, possibilitando que mais de uma energia característica de radionuclídeo seja encontrada no conjunto de amostras. Deve ser mantido, entretanto, em um valor que limite o número de centros de agrupamento em algumas unidades, para diminuir a descoberta de energias características não presentes nas amostras.

Os dados devem ser normalizados, ou seja, é assumido que os pontos de dados são delimitadas por uma unidade de hipercubo. A normalização pode ser feita com o maior valor do conjunto de amostras dividindo o módulo  $\|x_j - x_i\|$ , a distância euclidiana entre os pontos usados e  $x_j$ .

O valor potencial associado a cada um dos pontos depende da sua distância em relação a todos os seus vizinhos. Considerando a Equação 6, um ponto que tem muitas amostras em sua vizinhança terá um valor elevado de potencial, enquanto um ponto remoto terá um baixo valor de potencial. Depois de calcular o potencial de cada ponto, um a um, digamos que o ponto  $x_i^*$  possui o maior valor potencial  $P_i^*$ . Dessa forma  $x_i^*$  será selecionado como o primeiro centro de agrupamento. O potencial de cada ponto é então reduzido conforme definido pela Equação 7. Até que o critério de parada seja satisfeito, o algoritmo continua a seleção dos centros de agrupamento e a revisão dos potenciais iterativamente.

$$P_i = P_i - P_i^* e^{-\beta \|x_i - x_i^*\|^2}, \quad (7)$$

Na Equação 7,  $\beta = 4/r_b^2$  representa o raio de vizinhança para o qual uma revisão significativa de potencial irá ocorrer. Os pontos ou amostras que estão próximos do primeiro centro de agrupamento encontrado, no caso  $x_i^*$ , terão uma redução significativa no seu valor potencial. Logo, fazendo esses pontos improváveis para serem selecionados como o próximo centro de agrupamento. O valor de  $r_b$  deve ser igual ou maior que o valor de  $r_a$  para evitar centros de grupos muito próximos. O algoritmo 2 mostra os passos para a execução do método. Resumindo, as características do agrupamento subtrativo são:

1. O número de agrupamentos não é predefinido, sendo considerado *realmente* não supervisionado.
2. Com os cálculos recursivos, utilizando os mesmos dados, é computacionalmente de fácil administração e aplicável a problemas em tempo real.

**Algoritmo 2** Agrupamento Subtrativo

- 
- 1: **Entrada**  $x_i, x_j, r_a, r_b$ ;
  - 2: **Saída**  $P_i^*$  e  $x_i^*$ ;
  - 3: Usando a Equação 6, calcular o potencial  $P_i$  para cada ponto ou amostra  $x_i$ ,  $1 \leq i \leq n$ ;
  - 4: Selecionar o ponto  $x_i^*$  de maior valor potencial  $P_i^*$ ;
  - 5: **Repete**
  - 6: Revisar o valor potencial de cada ponto  $P_i$ , de acordo com a função, definida na Equação 7;
  - 7: Selecionar o novo ponto  $x_i^*$  de maior valor potencial  $MaxP_i$ ;
  - 8: **Até** O valor máximo encontrado  $MaxP_i \leq \epsilon P_i^*$ , onde  $\epsilon$  é o raio de rejeição.
- 

3. Pode começar “do zero” a partir da primeira amostra assumindo (temporariamente) o primeiro centro de um agrupamento;
4. Os agrupamentos obtidos pelo método podem ser usados para definir um conjunto de regras em métodos de agrupamentos nebulosos (FILEV; ANGELOV, 2007).

## 2.3 Testes com Agrupamento Subtrativo

O algoritmo de agrupamento subtrativo foi testado utilizando uma ferramenta disponível no ambiente MatLab chamada *subclust* (MATHWORKS, 1999). O objetivo destes testes foi verificar se amostras de valores de energia gama, captadas em detectores em presença de radionuclídeos, poderiam, ao passar pelo processamento realizado por este algoritmo, indicar agrupamentos correspondentes com esses valores de energia característica. Logo, um conjunto de dados com valores de energia, representados numericamente por valores de canais de um ADC, foram fornecidos como entrada para o algoritmo.

A Figura 8 mostra um espectro, gerado por simulação, que ilustra como esses pontos se distribuem. O eixo horizontal representa os canais para um ADC de 12 bits e o eixo vertical representa o número de vezes que cada valor de energia foi verificado neste conjunto de amostras. O espectro representa uma fonte radioativa composta por Césio-137 (Cs-137) e Cobalto-60 (Co-60). Nesta representação, 4096 canais correspondem a 2,048 MeV no espectro de energia. O primeiro pico no canal 1324 é característico do Cs-137 (0,662 MeV), enquanto o segundo, no canal 2340, e o terceiro no canal 2666 correspondem aos valores de energia característica do Co-60. São nestes picos que estão os agrupamentos que o algoritmo deve encontrar.

Os dados fornecidos pelo programa de simulação *Real Gamma-Spectrum Emulator* (KLEINRATH et al., 2010) são em forma de planilha de duas colunas, onde a primeira coluna

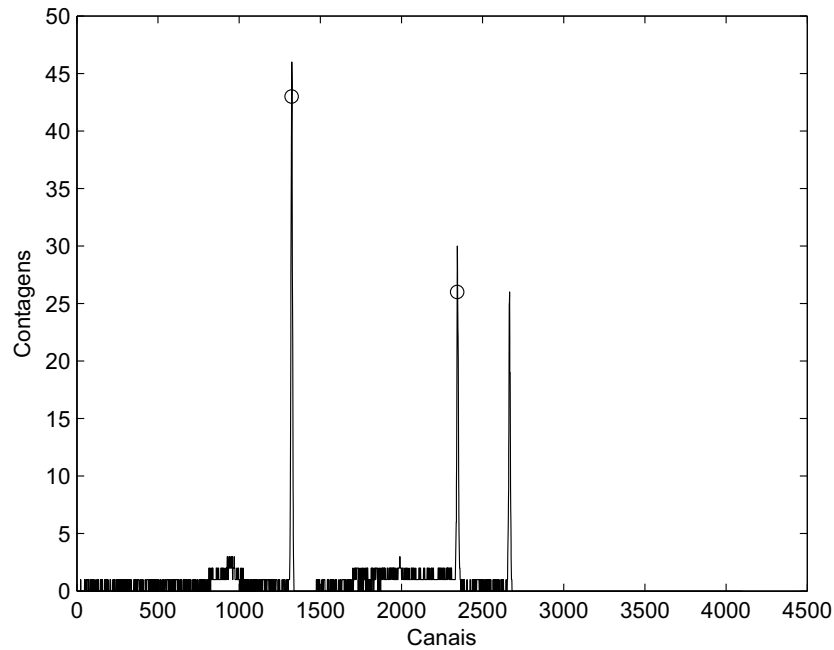


Figura 8: Espectro de energia simulado de uma fonte com Cs-137 e Co-60.

corresponde ao canal e a segunda ao número de ocorrências acumuladas em cada canal. Para aplicar o algoritmo de agrupamento subtrativo no MatLab, os dados fornecidos pelo programa de simulação precisam ser convertidos em dados unidimensionais, em uma só coluna. Por exemplo, se o canal 1324 acumulasse 100 ocorrências, significa que o valor 1324 deve aparecer 100 vezes como dado de entrada. Somente assim o algoritmo de agrupamento é capaz de perceber e dividir os dados de entrada em agrupamentos. Em uma aplicação real, estes dados equivaleriam à saída do ADC de um sistema de espectrometria gama, conforme explicado no Capítulo 1.

Na Figura 8, a marca circular no primeiro e segundo picos mostra o resultado da aplicação do algoritmo de agrupamento subtrativo. O centro do agrupamento encontrado é muito próximo (um canal para a esquerda) do resultado esperado. Com os parâmetros de configuração ajustados no algoritmo, o terceiro pico não foi encontrado. Este resultado pode ser alterado com um ajuste do raio  $r_a$  na Equação 6 e mais centros de agrupamento podem ser encontrados. Isso foi realizado em outros testes.

A identificação desses valores de energia na amostra é suficiente para permitir que esse resultado seja usado, em um processamento complementar, para descobrir que os dados fornecidos pertencem a uma fonte radioativa com Cs-137 e Co-60. Isso permite concluir que o método de agrupamento subtrativo pode ser usado para realizar a classificação desses valores

de energia em um sistema de identificação de radionuclídeos.

## 2.4 Considerações Finais do Capítulo

Este capítulo apresentou os conceitos de agrupamento e alguns métodos utilizados em várias atividades onde a classificação de dados é necessária. Foram apresentadas algumas justificativas para a escolha do algoritmo de agrupamento subtrativo para este trabalho.

O resultado apresentado na Seção 2.3 foi um dos testes realizados com o algoritmo sendo processado no MatLab. Estes mostraram que o método pode ser utilizado para a tarefa de classificar os valores de energia presentes nas amostras. E com ajustes nos parâmetros  $r_a$  e  $r_b$ , pode realizar os agrupamentos com boa precisão.

Após a verificação da viabilidade do uso deste método, o próximo capítulo inicia a apresentação da arquitetura de *hardware* desenvolvida para implementar este algoritmo de agrupamento subtrativo, descrevendo principalmente o controlador principal das unidades de cálculo.

## Capítulo 3

# MACRO-ARQUITETURA E MODELAGEM DOS DADOS

**O** PRESENTE capítulo aborda a arquitetura de *hardware* proposta neste projeto.

A Seção 3.1 fornece uma visão geral da arquitetura e contém informações sobre os objetivos gerais do *hardware* proposto. Os componentes principais que integram a macro-arquitetura são descritos na Seção 3.2. A Seção 3.3 apresenta uma visão geral da aproximação por polinômios, utilizada para calcular o valor exponencial  $e^{-\alpha\|x_j-x_i\|^2}$ , e da representação numérica dos dados de números reais em forma de frações de inteiros.

Na Seção 3.4 são apresentados os detalhes da unidade de controle do *hardware*, que também possui as memórias para carga e armazenamento dos dados. Esta seção mostra também os sinais do barramento de controle envolvidos na comunicação entre os componentes da macro-arquitetura de *hardware*. Na Seção 3.5 são apresentados os resultados da simulação para uma compreensão da atuação do controle da arquitetura. Na Seção 3.6 são apresentadas as considerações finais do capítulo e um resumo sobre o assunto tratado no Capítulo 4.

### 3.1 Visão Geral

O *hardware* desenvolvido neste projeto implementa o algoritmo de agrupamento subtrativo. O algoritmo de agrupamento subtrativo foi explicado no Capítulo 2. Este algoritmo calcula o valor potencial de cada ponto, definido como na Equação 8.

$$P_i = \sum_{j=1}^n e^{-\alpha\|x_j-x_i\|^2}, \quad (8)$$

onde  $\alpha = 4/r_a^2$ ,  $P_i$  é o valor potencial no ponto  $i$ ,  $x_i$  é o ponto atual e  $r_a$  é uma constante positiva, chamada de raio de agrupamento.

Após calcular o potencial de cada ponto ou amostra, uma a uma, o algoritmo seleciona o primeiro centro de agrupamento  $x_i^*$ , o ponto de maior valor potencial  $P_i^*$ . O potencial de cada ponto é então reduzido como definido na Equação 9.

$$P_i^{(t+1)} = P_i^{(t)} - P_i^* e^{-\beta \|x_i - x_i^*\|^2}, \quad (9)$$

onde  $P_i^{(t+1)}$  é o novo valor potencial calculado para o ponto  $i$ ,  $P_i^{(t)}$  é o valor potencial atual do ponto que está sofrendo redução e  $\beta = 4/r_b^2$  representa o raio de vizinhança para o qual uma significativa revisão de potencial irá ocorrer. Os pontos com maior proximidade do primeiro centro de agrupamento ( $x_i^*$ ) terão uma redução maior no valor potencial, tornando estes pontos menos prováveis a serem selecionados como o próximo centro de agrupamento. Como explicado no Capítulo 2, até que o critério de parada seja satisfeito, o algoritmo continua a seleção de novos centros de agrupamento e a revisão dos potenciais de cada ponto, iterativamente.

A Figura 9 mostra a macro-arquitetura, formada por duas unidades. Chamaremos de UAS, unidade de agrupamento subtrativo, a unidade onde se processa toda a computação aritmética, descrita anteriormente, para o cálculo dos potenciais de cada ponto no algoritmo de agrupamento subtrativo.

O outro componente desta macro-arquitetura será chamado de UCAC, unidade de carga, armazenamento e controle, que disponibiliza à UAS o conjunto de amostras para a seleção dos centros de agrupamentos e armazena os resultados dos potenciais calculados de cada amostra. Este componente possui também o controlador da UAS.

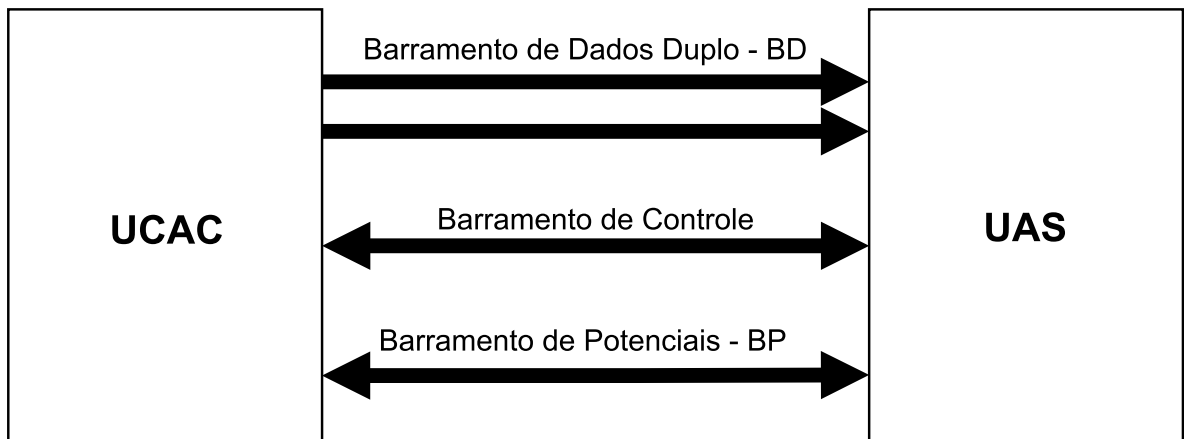


Figura 9: Diagrama da macro-arquitetura: UAS e UCAC

A UCAC, atuando como fornecedor e armazenador do conjunto de amostras, gerador dos sinais de controle e armazenador dos potenciais calculados, poderia ser um software em



um processador de propósito geral executando estas tarefas. Neste projeto a UCAC é um controlador, baseado em uma máquina de estados, também implementado em *hardware*. Esta opção se deveu ao fato desta configuração permitir maior rapidez e controle no fornecimento dos dados à UAS, uma vez que esta necessita iterativamente receber as amostras, que devem estar armazenadas em uma memória.

Como será visto nas seções seguintes, o componente UCAC possui duas memórias independentes, uma que armazena as amostras e outra que armazena os dados dos potenciais calculados de cada amostra. A memória que armazena as amostras possui duas saídas de dados. Por isso o barramento associado a ela é duplo, o barramento BD da Figura 9. A memória que armazena os dados dos potenciais calculados está associada ao barramento BP da Figura 9.

## 3.2 Componentes da Macro-Arquitetura

A unidade de agrupamento subtrativo (UAS) executa a computação aritmética necessária para a obtenção dos potenciais das amostras/pontos no processamento do algoritmo. Como mostrado na Figura 10, ela é formada por dois componentes, EXP1 e EXP2, que realizam a computação do valor exponencial  $e^{-\alpha\|x_j-x_i\|^2}$ , e o componente SOMADOR para realizar o somatório dos resultados de EXP1 e EXP2 no cálculo da Equação 8. Cada componente da UAS possui a sua unidade de controle independente.

Na Figura 10 são apresentados os sinais de controle entre a UCAC e a UAS. A UCAC atua como o elemento de controle principal do *hardware* proposto. Neste sentido ela é responsável, entre outras funções que serão descritas na próxima seção, por iniciar as atividades da UAS e de receber a informação desta quando sua atividade é concluída.

A função dos sinais de controle, envolvidos na comunicação entre a UCAC e a UAS, está descrita a seguir. Todos estes são sinais de 1 bit.

1. **IniciaExp**: Inicia, quando igual a 1, a execução nos componentes EXP da UAS.
2. **Se11**: Quando **Se11**=0, o multiplexador 1 seleciona o valor do registrador  $X_i$  para a entrada  $x_i$  dos componentes EXP1 e EXP2 da UAS. Esse valor será usado durante o cálculo do potencial no ponto  $i$ . Quando **Se1**=1 o valor selecionado para a entrada  $x_i$  é o valor do registrador  $X_{index}$ . Esse valor é o ponto de maior potencial, o primeiro centro de agrupamento  $x_i^*$ , utilizado na redução de potencial definida na Equação 9.
3. **LXi**: Na transição de 1 para 0 este sinal carrega no registrador  $X_i$  o valor  $x_i$  que será usado pelos componentes EXP1 e EXP2 da UAS no cálculo do valor potencial neste ponto.

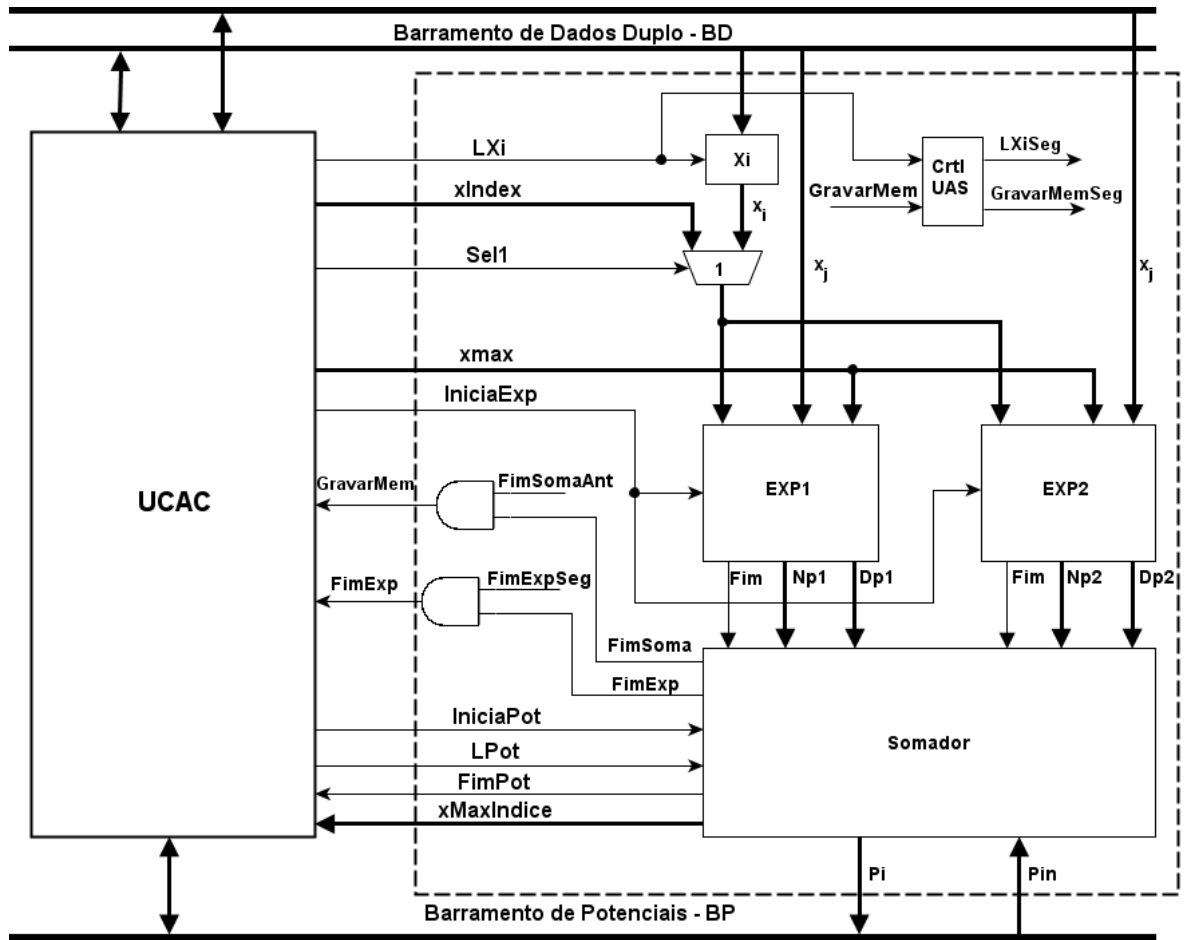


Figura 10: Unidade de agrupamento subtrativo - UAS

4. **LXiSeg**: Sinal para carregar o  $X_i$  do componente UAS seguinte, caso exista.
5. **LPot**: Na transição de 1 para 0 este sinal carrega dois registradores internos do componente SOMADOR da UAS com valores de potencial máximo (numerador e denominador) para uso durante a redução de potencial.
6. **FimExp**: Este sinal da UAS para a UCAC informa ao controle principal que o cálculo de um fator do somatório, dado pela Equação 8, foi concluído nos dois componentes EXP.
7. **FimExpSeg**: Em um arquitetura com mais de um componente UAS este sinal informa que no UAS seguinte os dois componentes EXP terminaram. O último componente UAS recebe este sinal igual a 1.
8. **FimSoma**: Informa que a última soma ocorreu e o somatório da Equação 8, para o cálculo do potencial em um ponto, terminou.

9. **FimSomaAnt**: Em uma arquitetura com mais de um componente UAS este sinal informa que o UAS anterior terminou o somatório e gravou o resultado.
10. **GravarMem**: Este sinal informa que o dado em  $P_i$ , o potencial calculado, pode ser gravado na memória MP, a memória de potenciais interna da unidade UCAC. Para uma arquitetura com somente um componente UAS este sinal é equivalente ao sinal **FimSoma** pois o sinal **FimSomaAnt** é igual a 1.
11. **GravarMemSeg**: Sinal que informa à UAS seguinte a possibilidade de gravação dos dados.
12. **IniciaPot**: Este sinal informa para a UAS iniciar o cálculo para a redução de potencial, definida pela Equação 9. O controle do componente SOMADOR recebe este sinal e executa uma parte distinta de sua máquina de estados para realizar o cálculo definido por esta equação.
13. **FimPot**: Informa que o processo de redução de potencial foi terminado.

Baseando-se na Equação 8 e nas explicações do Capítulo 2, o componente UAS precisa, para calcular o potencial no ponto  $i$ , do valor numérico  $x_i$  e do valor numérico de todos os pontos do conjunto de amostras ( $x_j$ ) para realizar o somatório. Cada componente EXP realiza a computação de um fator ( $e^{-\alpha \|x_j - x_i\|^2}$ ) do somatório da Equação 8, sendo calculado dois fatores em paralelo por vez. Desta forma, cada valor  $x_i$  pode ser registrado e fornecido aos componentes EXP1 e EXP2 da UAS durante o cálculo do potencial no ponto  $i$ . Isto é feito no registrador Xi que disponibiliza o valor através do multiplexador 1.

Um novo valor  $x_i$  é registrado a partir da memória MD para calcular o potencial da amostra subsequente. Isto ocorre depois que todos os valores  $x_j$  do conjunto de  $n$  amostras forem fornecidos à UAS e o somatório da Equação 8, ou seja, o potencial no ponto  $i$ , tenha sido calculado. Os valores  $x_j$  são fornecidos à UAS pela UCAC através do barramento duplo BD que pode ser visto na Figura 10.

A Figura 10 mostra que o resultado do cálculo nos componentes EXP1 e EXP2 são as saídas  $N_p$  e  $D_p$  (numerador e denominador), que representam as frações que serão usadas no SOMADOR. Cada componente EXP informa também ao SOMADOR o término da operação pelo sinal de saída **Fim**.

A UCAC armazena os dados dos potenciais calculados de cada amostra, gravando-os a partir da saída  $P_i$  do SOMADOR. Posteriormente a UCAC fornece esses valores à entrada  $P_{in}$ , através do barramento de potenciais BP, para a revisão dos potenciais definida na Equação 9.

A arquitetura proposta permite que a unidade de agrupamento subtrativo (UAS) seja replicada, acrescentando outros destes componentes em paralelo para a computação dos fatores  $e^{-\alpha\|x_j-x_i\|^2}$ . Isto fornece maior flexibilidade à implementação do *hardware*, permitindo que se avalie o ganho de desempenho em função da área utilizada. A escalabilidade do *hardware* será vista em mais detalhes no Capítulo 4.

### 3.3 Aproximação por Polinômios e Representação numérica

O objetivo desta seção é entender as escolhas feitas neste trabalho pela aproximação por polinômios e pela representação numérica de números reais em forma de frações de inteiros, escolhas estas que se refletiram na definição da arquitetura para o cálculo e somatório do valor exponencial  $e^{-\alpha\|x_j-x_i\|^2}$ , desempenhados pelo componente UAS.

#### 3.3.1 Representação numérica

Uma escolha que se refletiu na definição da arquitetura foi a representação numérica de números reais em forma de frações de inteiros. A principal razão desta escolha foi buscar a simplificação do *hardware*, considerando que uma perda de precisão nos cálculos não afetaria o encontro do resultado correto.

Visto que o *hardware* apresenta tamanho limitado em bits para a representação de numerador e denominador, é sabido que quando um número real é convertido numa fração de inteiros haverá uma perda de precisão. Portanto, é importante que a conversão busque a fração que melhor se aproxime do número real pretendido. Alguns algoritmos, como o inspirado no fluxograma de “força bruta” (*Brute Force*) (SANTI-JONES; GU, 2008) são utilizados para este fim.

A representação numérica utilizada é baseada no modelo *Fractional Fixed Point* (FFP) (SANTI-JONES; GU, 2008). Com este modelo é possível trabalhar com operações em números reais sem usar o formato de ponto flutuante (IEEE Floating Point) (TANENBAUM, 2007). A representação e as operações matemáticas (soma e multiplicação) em ponto flutuante são mais complexas e tendem a demandar mais área na implementação em *hardware* digital. Com um número real tratado como fração, somente são utilizadas operações com números inteiros, o que torna o *hardware* mais simples.

Portanto, no cálculo de cada valor  $e^{-v}$  pelo *hardware*, onde  $v = \alpha\|x_j-x_i\|^2$ , o valor  $v$  será tratado não como um número real, mas com uma fração de inteiros. Sendo  $\alpha$  uma constante

com valor inteiro,  $x_j$  e  $x_i$  também valores inteiros, o valor  $v$  também é um inteiro. Porém, os dados que o algoritmo de agrupamento subtrativo processa precisam ser normalizados, como visto no Capítulo 2. Para que sejam normalizados entre 0 e 1, neste caso, basta que o maior valor do conjunto de amostras ( $x_{max}$ ) seja usado para dividir o valor  $||x_j - x_i||$ . Assim o valor  $v$ , do qual será calculado o fator  $e^{-v}$ , passa a ser uma fração.

$$v = \frac{Nv}{Dv} = \frac{\alpha ||x_j - x_i||^2}{(x_{max})^2} \quad (10)$$

A fração definida na Equação 10 é montada dentro dos componentes para cálculo do valor exponencial (EXP1 e EXP2) que recebem os valores  $x_j$  e  $x_i$  pelo barramento de dados duplo (BD), registrando o valor  $x_i$ . O valor  $x_{max}$ , que não é alterado durante o cálculo, é registrado na UCAC, também a partir do barramento BD. O barramento de dados BD fornece números inteiros com um número de bits  $n$  suficiente para manipular o conjunto de dados fornecidos, considerando que quanto maior o número de bits menor é a perda de precisão.

Qualquer operação de soma, subtração ou multiplicação com frações resulta numa outra fração, que pode ser representada na notação usada nesta arquitetura (SILVA; NEDJAH; MOURELLE, 2009): numerador natural e denominador natural diferente de zero. Considerando  $a$  e  $b$  números reais, exemplifica-se abaixo algumas operações.

$$\begin{aligned} \text{Soma :} \quad a + b &\mapsto \frac{Na}{Da} + \frac{Nb}{Db} = \frac{Na \times Db + Da \times Nb}{Da \times Db} = \frac{\text{Natural}}{\text{Natural diferente de zero}} \\ \text{Subtração :} \quad a - b &\mapsto \frac{Na}{Da} - \frac{Nb}{Db} = \frac{Na \times Db - Da \times Nb}{Da \times Db} = \frac{\text{Natural}}{\text{Natural diferente de zero}} \\ \text{Multiplicação :} \quad a \times b &\mapsto \frac{Na}{Da} \times \frac{Nb}{Db} = \frac{Na \times Nb}{Da \times Db} = \frac{\text{Natural}}{\text{Natural diferente de zero}} \end{aligned} \quad (11)$$

Os resultados das operações citadas podem ser representados em forma de fração. Porém, nessas operações, tanto o numerador quanto o denominador podem exceder o tamanho em bits manipulado pelo *hardware*. Em uma operação de multiplicação entre frações, por exemplo, o numerador  $Na \times Nb$  irá apresentar  $2n$  bits de maneira generalizada, uma vez que  $Na$  e  $Nb$  possuam  $n$  bits cada um. Logo no *hardware* proposto, qualquer fração que resulta da operação de outras frações deve ter o seu numerador e denominador ajustados para o número  $n$  de bits usado, antes da próxima operação. Este processo, que chamaremos de **Enquadramento**, faz parte do *hardware* dos componentes EXP e SOMADOR da UAS e será explicado em mais detalhes no Capítulo 4.

### 3.3.2 Aproximação por polinômios

No Capítulo 2 foi visto que a implementação do algoritmo de agrupamento subtrativo busca obter o ponto de maior potencial calculado dentro de um conjunto de amostras. O *hardware* para esta implementação, portanto, necessita calcular estes valores com uma precisão adequada tão somente a compará-los e poder distinguir o de maior valor. A resposta buscada pelo algoritmo é o índice, no conjunto de amostras, que corresponde ao maior valor, correspondente ao centro do agrupamento. Portanto, o cálculo de cada fator  $e^{-v}$  do somatório da Equação 8 admite uma aproximação.

Neste trabalho, o valor exponencial  $e^{-v}$  é computado, através de operações aritméticas, usando aproximação a polinômios quadráticos pelo Método dos Mínimos Quadrados (MMQ) (RUGGIERO; LOPES, 1988). Desta forma, um polinômio quadrático é ajustado à exponencial  $e^{-v}$  num intervalo definido de  $v$ . Cada polinômio  $P(v) = Av^2 + Bv + C$  tem seus coeficientes ( $A$ ,  $B$  e  $C$ ) ajustados para se aproximar de  $e^{-v}$  numa faixa de  $v > 0$ . Quanto menor a faixa, menor é o erro da aproximação por polinômios. Neste caso foram escolhidas cinco faixas para aproximação:  $v \in [0, 1[$ ,  $v \in [1, 2[$ ,  $v \in [2, 4[$ ,  $v \in [4, 8[$  e  $v \in [8, +\infty[$ . Estes coeficientes são mostrados respectivamente, já em forma de fração, nos polinômios quadráticos da Equação 13. O valor  $Nv/Dv$  é equivalente a  $v$ . Os coeficientes  $Na/Da$ ,  $Nb/Db$  e  $Nc/Dc$  são equivalentes a  $A$ ,  $B$  e  $C$  e estão representados conforme Equação 12.

$$e^{-\frac{Nv}{Dv}} = \frac{Na}{Da} \left( \frac{Nv}{Dv} \right)^2 + \frac{Nb}{Db} \left( \frac{Nv}{Dv} \right) + \frac{Nc}{Dc} \quad (12)$$

$$e^{-\left(\frac{Nv}{Dv}\right)} \cong \begin{cases} P_{[0,1[}\left(\frac{Nv}{Dv}\right) = \frac{773}{2500} \left(\frac{Nv}{Dv}\right)^2 - \frac{372}{400} \left(\frac{Nv}{Dv}\right) + \frac{9953}{10000} \\ P_{[1,2[}\left(\frac{Nv}{Dv}\right) = \frac{569}{5000} \left(\frac{Nv}{Dv}\right)^2 - \frac{2853}{5000} \left(\frac{Nv}{Dv}\right) + \frac{823}{1000} \\ P_{[2,4[}\left(\frac{Nv}{Dv}\right) = \frac{67}{2500} \left(\frac{Nv}{Dv}\right)^2 - \frac{2161}{10000} \left(\frac{Nv}{Dv}\right) + \frac{4565}{10000} \\ P_{[4,8[}\left(\frac{Nv}{Dv}\right) = \frac{16}{10000} \left(\frac{Nv}{Dv}\right)^2 - \frac{234}{10000} \left(\frac{Nv}{Dv}\right) + \frac{835}{10000} \\ P_{[8,\infty[}\left(\frac{Nv}{Dv}\right) = 0 \end{cases} \quad (13)$$

Para  $v = 8$ , tem-se  $e^{-8} \cong 0,0003355$ . Neste caso, considera-se válida a seguinte aproximação  $e^{-v} \Big|_{v \geq 8} \cong 0$ . Dessa maneira, se estabelece o polinômio nulo em  $P_{[8,\infty[}$ .

Os coeficientes do polinômio quadrático apresentados devem ser guardados e fornecidos nos componentes EXP. Esses dados são previamente calculados e armazenados em cada compo-

nente EXP utilizado na arquitetura, com o mesmo número de bits  $n$  escolhido para numerador e denominador. No Capítulo 4 será mostrado esse circuito.

A Figura 11 mostra a exponencial natural  $e^{-v}$ . Na Figura 12 estão apresentadas as aproximações das curvas em cada intervalo, utilizando os coeficientes conforme Equação 13.

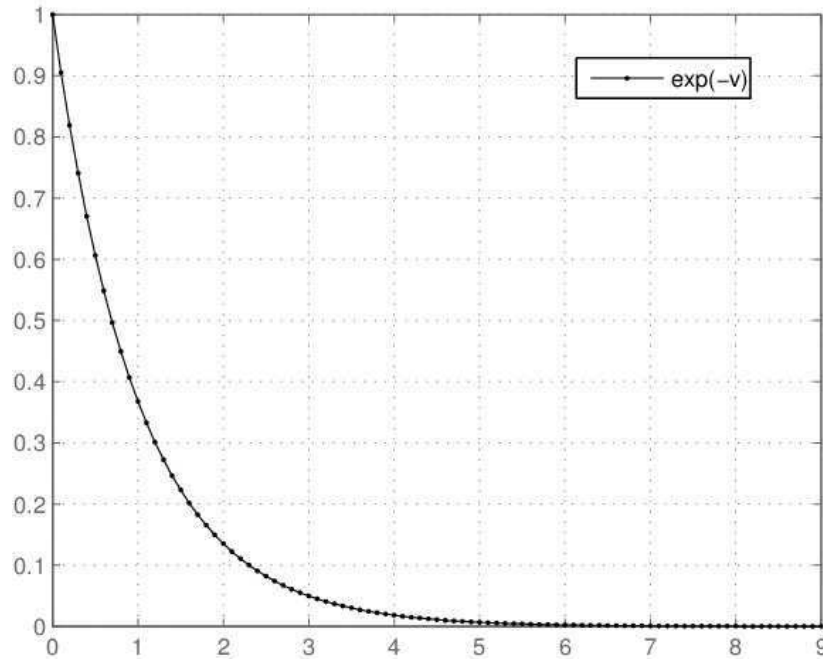


Figura 11: Exponencial natural  $e^{-v}$  para  $v \geq 0$

O erro encontrado nesta aproximação é mostrado na Figura 13. O intervalo  $[0, 1[$  apresenta erros mais acentuados, mas para este projeto esses valores são satisfatórios.

### 3.4 Unidade de Carga, Armazenamento e Controle

A unidade de carga, armazenamento e controle (UCAC) é mostrada na Figura 14. Esta possui duas memórias independentes. Uma memória dupla (MD), que armazena as amostras, e outra memória que armazena os dados dos potenciais calculados de cada amostra (MP). A memória MD permite que os dois componentes (EXP1 e EXP2) da UAS recebam ao mesmo tempo uma amostra para o cálculo do valor exponencial e deste modo possam operar em paralelo. Esta amostra para cada componente EXP são dois valores  $x_j$  distintos de dois endereços subsequentes da memória.

O circuito para fornecer o endereço à memória MD é formado por dois multiplexadores e dois registradores com entrada para incremento. O registrador AddXi armazena o endereço de memória para fornecer o valor  $x_i$  no cálculo do valor exponencial. O valor potencial calculado

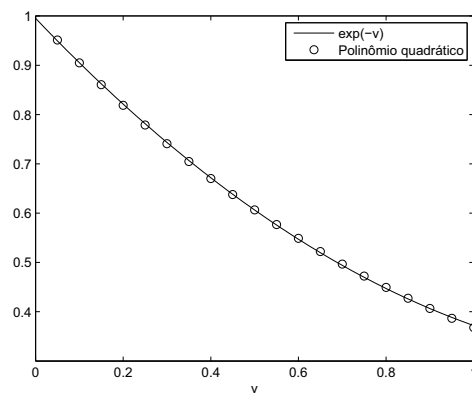
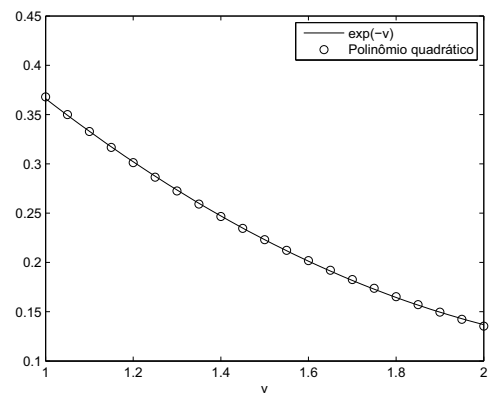
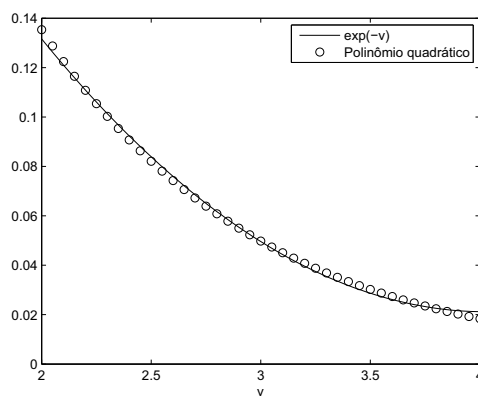
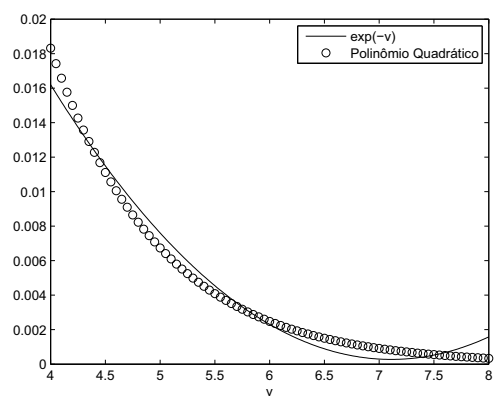
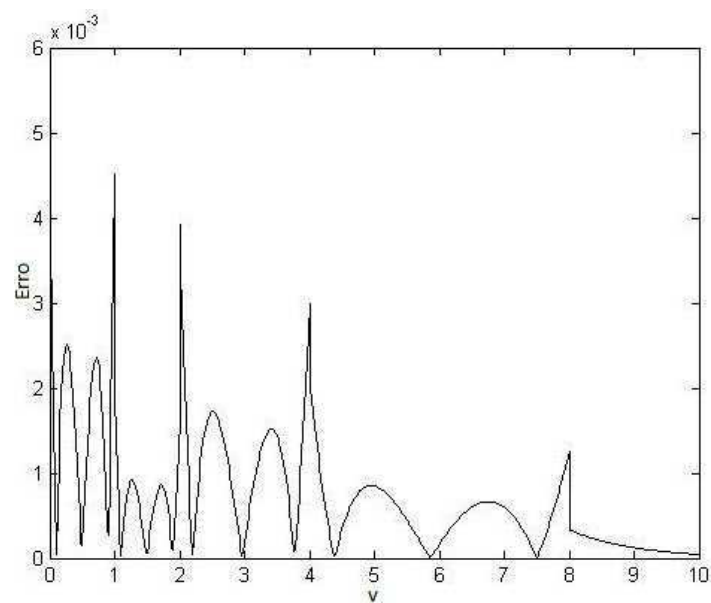
(a) Ajuste de curva em  $[0, 1]$ :(b) Ajuste de curva em  $[1, 2]$ :(c) Ajuste de curva em  $[2, 4]$ :(d) Ajuste de curva em  $[4, 8]$ :Figura 12: Polinômios quadráticos aproximadores a  $e^{-v}$ , em  $[0, 8]$ 

Figura 13: Erro absoluto da aproximação por polinômios



corresponde ao ponto de valor  $x_i$ , cujo endereço é fornecido por este registrador e armazenado no registrador Xi do componente UAS.

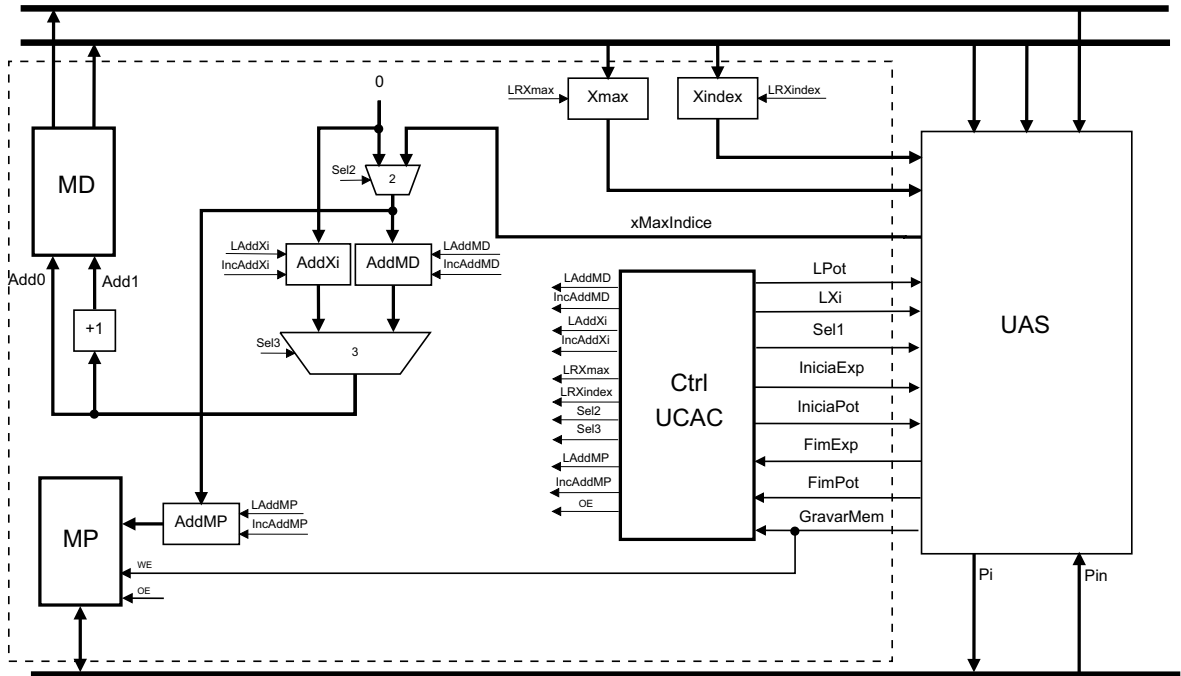


Figura 14: Unidade de Carga, Armazenamento e Controle - UCAC

O registrador *AddMD* armazena o endereço de memória para fornecer o valor  $x_j$  no cálculo do valor exponencial. Como a entrada do segundo endereço da memória dupla *MD* é incrementada em uma unidade em relação a entrada do primeiro endereço, o barramento de dados duplo (BD) fornece dois endereços subsequentes da memória aos componentes *EXP1* e *EXP2* da UAS, permitindo, como já visto, o cálculo de dois fatores  $e^{-v}$  em paralelo.

O sinal 0 na entrada do multiplexador 2 fornece o valor zero com  $n$  bits para inicializar o registrador *AddMD* quando necessário. O valor *xMaxIndice* é o índice do primeiro centro de agrupamento  $x_i^*$ , valor de potencial máximo usado na redução de potenciais mostrada pela Equação 9.

Para fornecer o endereço à memória *MP* é utilizado somente o registrador com incrementador *AddMP*, que recebe como entrada a saída do multiplexador 2, assim como o registrador *AddMD*. Estes registradores funcionam com dois sinais de controle da seguinte forma: quando o sinal de carregamento e incremento são ativados pelo controlador, o seu valor é incrementado. Quando somente o sinal de carregamento é ativado, o registrador carrega o valor da entrada.

A UCAC possui também o controlador principal do *hardware* proposto neste projeto. Os sinais de entrada e saída do controlador, para controle do endereçamento das memórias

(barramento de controle interno) e para controle do componente UAS (barramento de controle externo), são mostrados na Figura 14. Este controle tem como funções principais:

- Disponibilizar os valores  $x_j$ , armazenados na memória dupla MD, aos componentes EXP, através do barramento duplo BD;
- Iniciar os vários ciclos de cálculo do valor exponencial até que os dados em memória sejam computados e somados;
- Iniciar o cálculo da redução de potencial de cada ponto;

A máquina de estados deste controlador é mostrada na Figura 15. A descrição de cada estado é mostrada a seguir. Na primeira parte é realizado o controle para o cálculo do potencial máximo, o primeiro centro de agrupamento, descrito no Capítulo 2. Na segunda parte o controle atua para efetuar a revisão de potencial em cada ponto. Este controle já inclui estados utilizados para que a arquitetura disponha de mais componentes UAS em paralelo. Essa escalabilidade da arquitetura será explicada na Seção 4.4.

- S0: Primeira parte. Se sinal *Início* é apresentado vai para o estado S1;
- S1: Carrega o registrador de endereço da memória MD (AddMD) e o registrador AddXi com o valor inicial 0.  
Prepara para o carregamento do registrador XMAX (usado para efetuar a normalização) no próximo estado e vai para S2;
- S2: Prepara o incremento do registrador AddXi no próximo estado e também o carregamento do registrador AddMD com zero. Vai para S3;
- S3: Prepara para o incremento do registrador AddMD no próximo estado;  
Se o valor em AddXi é maior que o número de pontos na memória MD, vai para S14 para iniciar a redução de potenciais. Do contrário vai para S4;
- S4: Carrega o registrador Xi com valor da memória no endereço dado por AddXi.  
Se não existem mais componentes UAS (NumUAS=0) para receber outro valor em seu registrador Xi, vai para S5. Do contrário vai para S6;
- S5: Emite sinal para reiniciar, no próximo estado, o contador de UAS com o número destes componentes existentes na arquitetura. Vai para S9;

- S6: Se o valor em AddXi é maior que o número de pontos na memória MD, vai para S9. do contrário retorna para S4, emitindo sinais para incrementar AddXi e decrementar o contador do número de componentes UAS neste próximo estado;
- S7: Prepara para o incremento do registrador AddMD no próximo estado e vai para S8;
- S8: Prepara para o incremento do registrador AddMD no próximo estado e vai para S9;
- S9: Inicia os componentes EXP (sinal **IniciaExp** ) em todos as unidades UAS existentes e vai para S10;
- S10: Se os componentes EXP terminaram (sinal **FimExp**=1), vai para S11;
- S11: Se o endereço da posição de memória MD (AddMD) é final (**FimMD**=0), vai para S12; do contrário, vai para S13;
- S12: Sinaliza o fim de dados na memória MD e vai para S2;
- S13: Sinaliza que existem mais dados na memória MD e vai para S7;
- S14: Segunda parte - Redução de potenciais. Aguarda o sinal de finalização da última soma no componente UAS (**GravarMem**=1). Se terminou vai para S15;
- S15: Prepara o carregamento do registrador AddMD com o valor xMaxIndice no próximo estado. xMaxIndice é o índice do primeiro centro de agrupamento ( $x_i^*$ ) calculado. Vai para S16;
- S16: Prepara o carregamento do registrador Xindex, no próximo estado, com o valor do ponto de maior índice potencial encontrado ( $x_i^*$ ); vai para S17;
- S17: Prepara o carregamento do registrador AddXi com valor 0 no próximo estado e vai para S18;
- S18: Prepara o carregamento dos registradores AddMD e AddMP com 0 no próximo estado; vai para S19;
- S19: Carrega os registradores AddMD e AddMP com 0 e vai para S20;
- S20: Prepara o incremento do registrador AddMD e o registrador do valor potencial máximo, o valor  $P_i^*$  da Equação 9 (usado em todo o processo de redução) no próximo estado. vai para S21;

- S21: Inicia os componentes EXP (sinal **IniciaExp** );  
Se o endereço da posição de memória de MD (**AddMD**) é final (**FimMD**=0), vai para S26;  
do contrário, vai para S22;
- S22: Prepara o incremento do registrador **AddMD** no próximo estado. Vai para S23;
- S23: Incrementa o registrador **AddMD**. Vai para S24;
- S24: Torna igual a 1 o sinal **IniciaPot** para sinalizar à UAS que a soma seguinte será conforme a Equação 9.  
Se os componentes EXP terminaram (sinal **FimExp**), vai para S25;
- S25: Torna igual a 0 o sinal **IniciaPot**. Retorna para S21;
- S26: Verifica se nova redução de potencial deve ocorrer (sinal **FimPot** = 0). Se não, termina e retorna para S0. Se sim, retorna para S14.

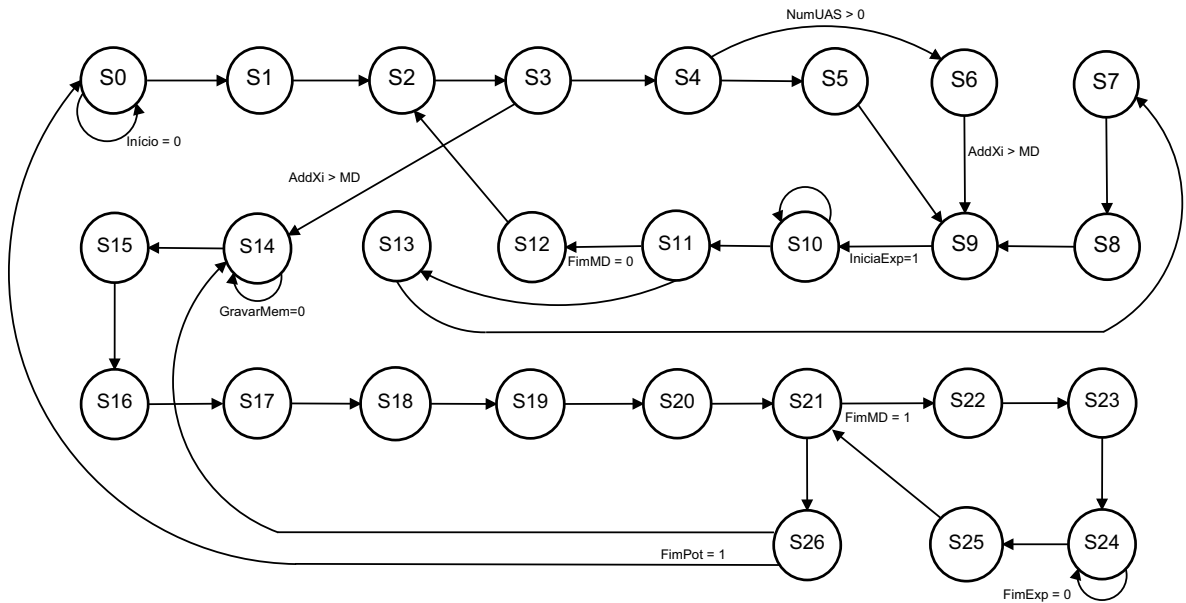


Figura 15: Máquina de estados do controlador principal

### 3.5 Simulações

Esta Seção apresenta janelas de simulação da arquitetura que visam demonstrar a atuação dos principais sinais de controle para a execução do algoritmo. O simulador utilizado foi o *Modelsim* (MENTOR-GRAPHICS, 2011). A simulação foi realizada com um conjunto de dados (valores de

1 a 10) armazenados na memória MD nos endereços 1 a 10. O endereço 0 da memória MD armazena o valor  $x_{max}$ , usado para efetuar a normalização dos dados.

A Figura 16 mostra o início do processamento. Os passos podem ser acompanhados em mais detalhes na máquina de estados descrita na Seção 3.4. Com o sinal **Início** apresentado, carrega-se o registrador **Xmax** com o valor do endereço 0 da memória MD. Neste caso o valor 15. Logo em seguida o endereço da memória MD (**Add0** e **Add1**) é incrementado e o barramento duplo **BD** passa a fornecer os valores dos endereços 1 e 2, neste caso os valores 1 e 2 respectivamente, como pode ser visto em **BD-0** e **BD-1** da Figura. Estes são os dois valores  $x_j$  que serão usados pelos dois componentes **EXP** da **UAS** para calcular o valor exponencial  $e^{-v}$ , onde  $v$  é dado pela Equação 10. No passo seguinte, com o sinal **LXi**, o valor  $x_i$  é carregado no registrador **Xi** a partir de **BD-0**.

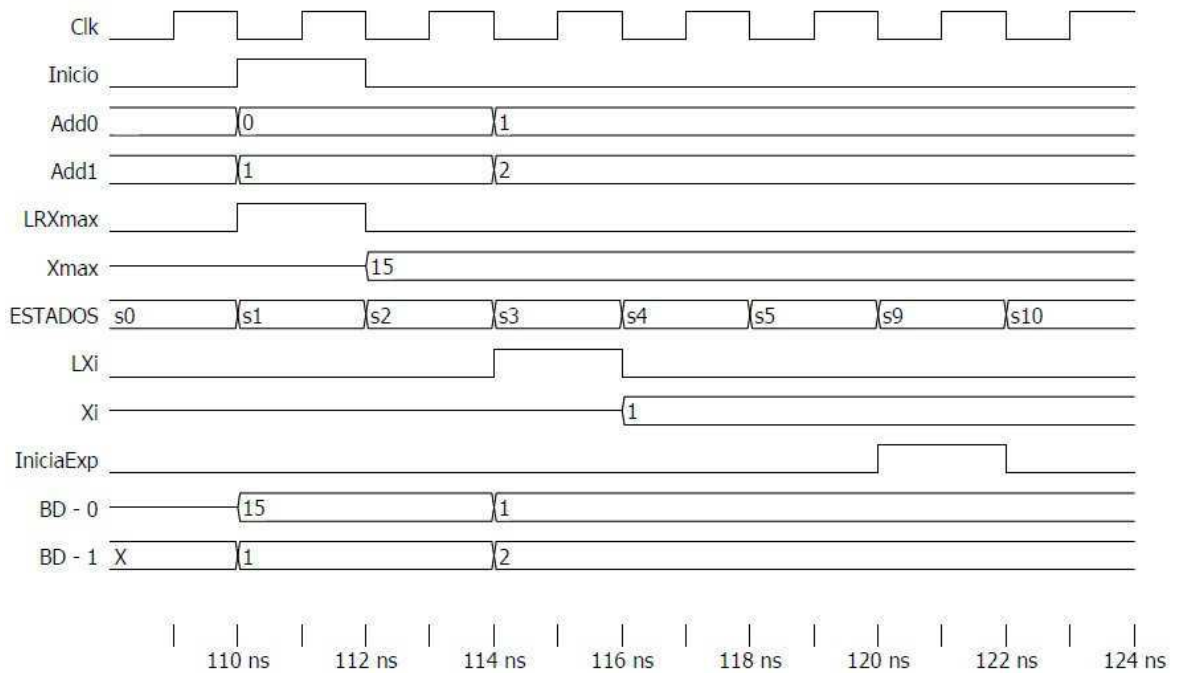


Figura 16: Início da simulação

Nos estados seguintes o controle verifica se existem outros componentes **UAS** na arquitetura. Caso existissem, valores  $x_i$  subsequentes na memória já seriam disponibilizados. Isto será visto quando explicarmos a possibilidade de escalabilidade da arquitetura na Seção 4.4 do Capítulo 4. Em **S9** o cálculo nos componentes **EXP** é iniciado com o sinal **IniciaExp** do controle.

A Figura 16 apresenta a simulação até o estado S10 da máquina de estados descrita na Seção 3.4. Neste estado o fim do cálculo nos dois componentes EXP é aguardado.

A Figura 17 mostra a simulação a partir do momento em que ocorre o fim do cálculo nos componentes EXP. Neste caso os dois componentes terminaram o cálculo no mesmo instante. Com os dois sinais **FimExp** em 1, o sinal **Inicio** no componente SOMADOR também vai a 1 e a soma dos valores calculados em EXP1 e EXP2 é iniciada.

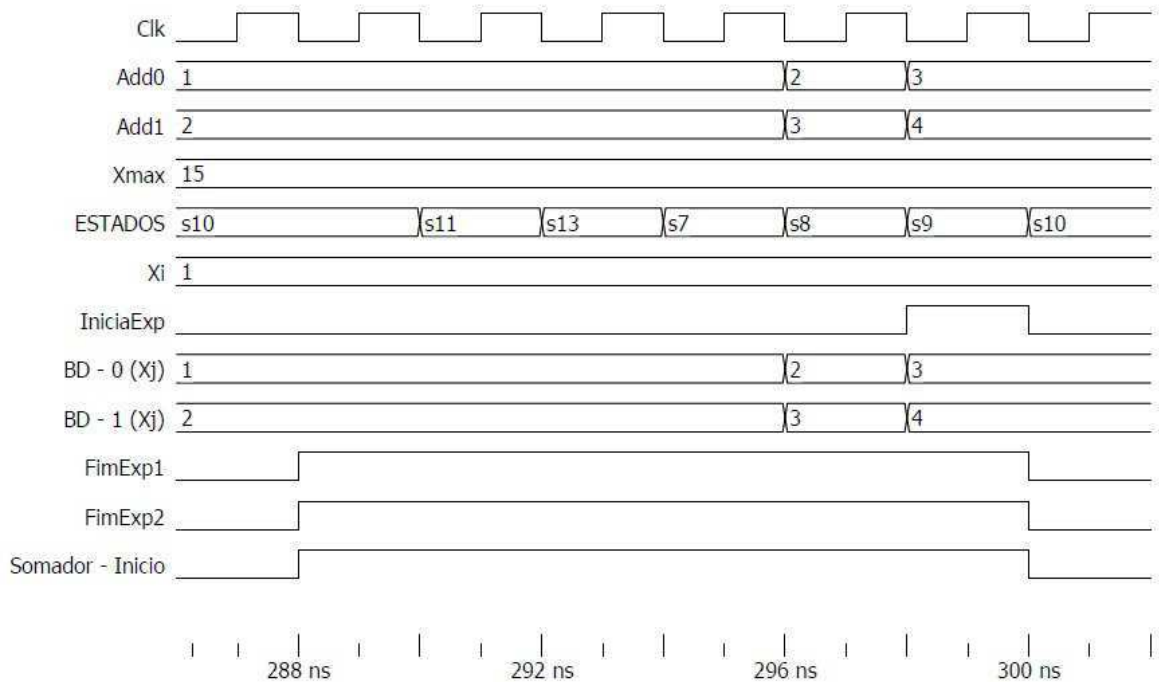


Figura 17: Simulação no fim do cálculo nos componentes EXP

O controle verifica no passo seguinte se existem mais dados na memória MD, ou seja, se todos os valores  $x_j$  foram computados. Isso é verificado pelo valor do endereço da memória MD. Se não terminaram, o endereço é incrementado duas vezes para disponibilizar no barramento BD os dois valores  $x_j$  subsequentes. No caso os valores 3 e 4. Os valores de  $x_i$  e  $x_{max}$  são mantidos.

Ocorre então o retorno ao estado S9, que inicia novamente o cálculo nos dois componentes EXP. Neste momento, tanto os componentes EXP quanto o componente SOMADOR estão operando. A simulação mostrando os cálculos nestes componentes será mostrada no Capítulo 4.

A Figura 18 mostra uma situação mais adiante, no processamento dos valores  $x_j$  dos endereços 5 e 6, onde o final do cálculo nos dois componentes EXP não ocorre no mesmo

instante. Nota-se que o sinal **Início** no componente SOMADOR só ocorre quando os dois sinais **FimExp** estão ativos.

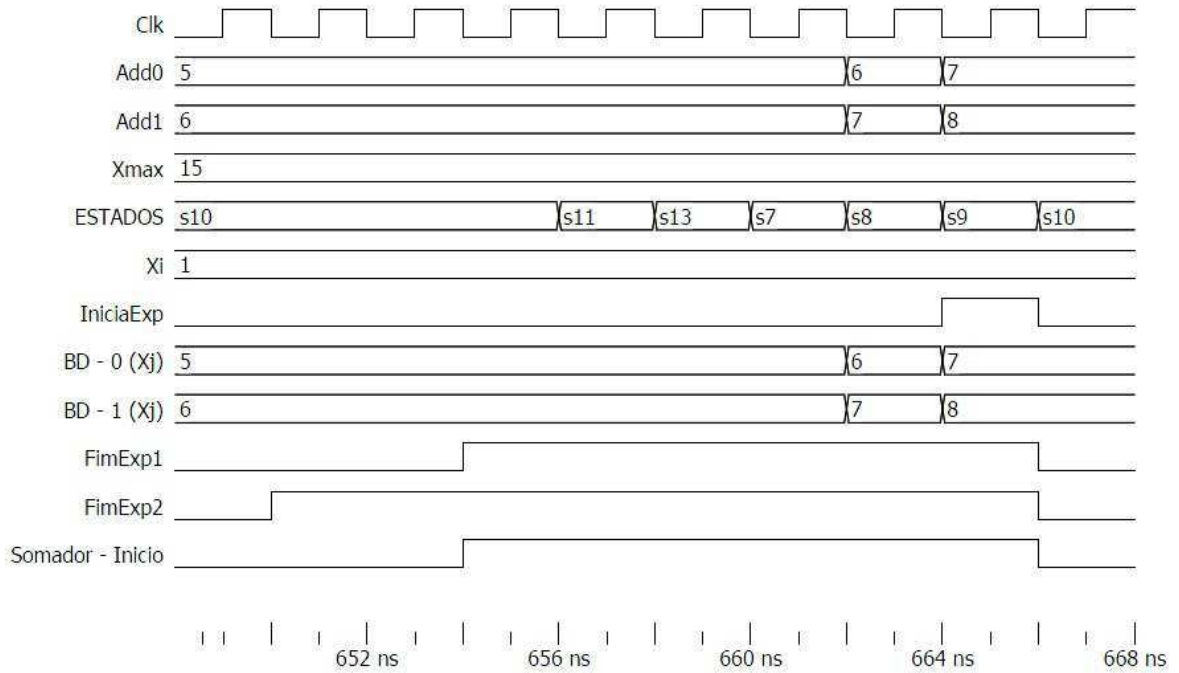


Figura 18: Simulação no fim do cálculo nos componentes EXP nos endereços 5 e 6

O somatório continua até que se chegue aos endereços 9 e 10 na memória MD, caso em que todos os valores  $x_j$  foram computados. Esta situação é mostrada na Figura 19.

Com todos os valores  $x_j$  calculados, há um desvio do estado S11 para o estado S12. O estado seguinte (S2) prepara o incremento do registrador AddXi, que guarda o endereço da memória MD do ponto  $x_i$ , o ponto cujo o potencial está sendo calculado no somatório. O estado S3 verifica se este ponto  $x_i$  é o último. Se não for, o estado seguinte carrega o novo valor no registrador Xi, como mostrado na Figura 19 em S4. Depois os valores  $x_j$  voltam a ser 1 e 2 e inicia-se novamente o cálculo nos componentes EXP.

O componente SOMADOR, ao terminar a soma iniciada nesta janela de simulação, indica o evento com o sinal **FimSoma**. Isto é mostrado na Figura 20. Neste momento o valor do potencial do primeiro ponto  $x_i$  é guardado na memória MP no endereço 0. A Figura mostra o momento da gravação com o sinal para habilitar a escrita (**WE-MP**) sendo apresentado no mesmo instante. A metade menos significativa do dado, neste caso de 32 bits, corresponde ao denominador da fração resultado, com 16 bits. A metade mais significativa corresponde ao numerador.

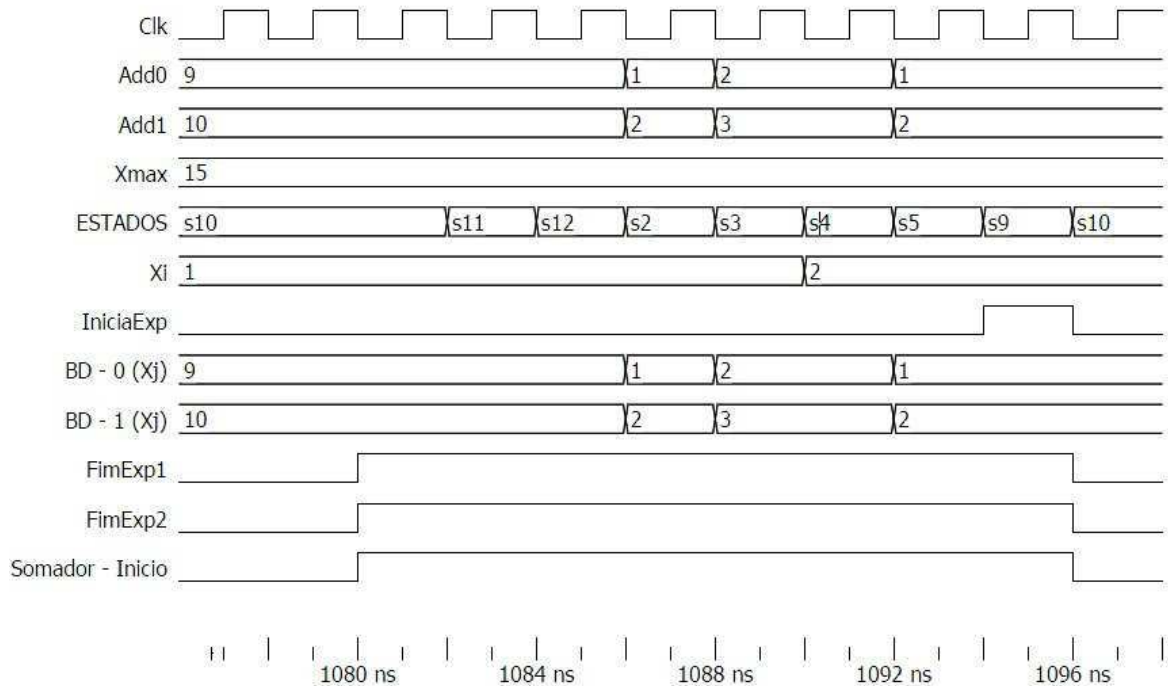


Figura 19: Simulação no fim do cálculo nos componentes EXP nos endereços 9 e 10

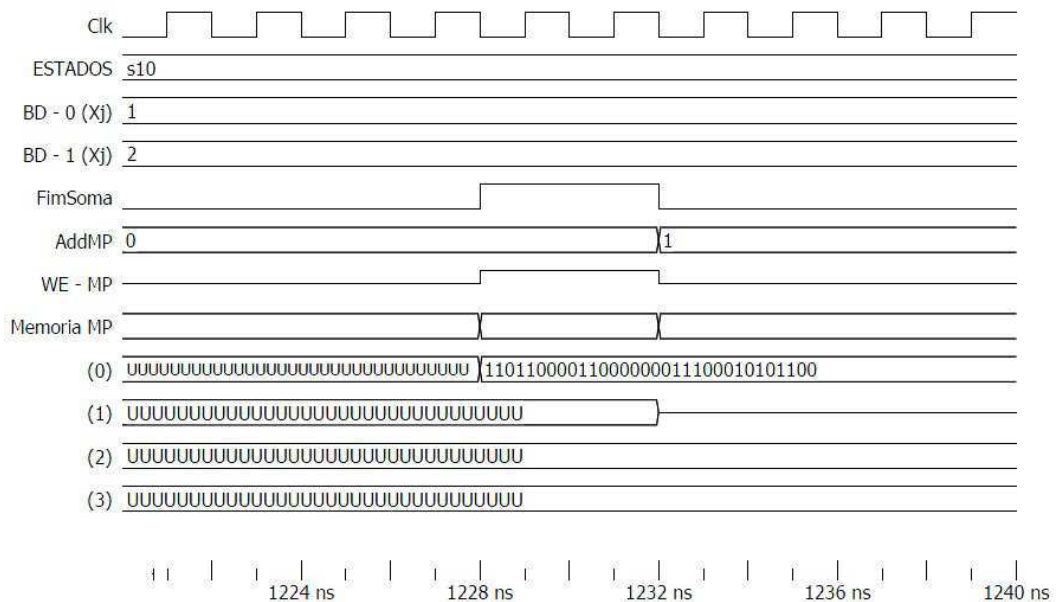


Figura 20: Simulação no fim do somatório e gravação dos dados

A medida que o valor potencial de cada ponto é calculado, o valor é gravado na memória MP em endereço subsequente. Quando todos os pontos da memória MD estiverem com o seu potencial calculado, inicia-se a redução de potencial conforme a Equação 9. A Figura 21 mostra



este momento, que começa no estado S14 da máquina de estados do controlador, descrita na Seção 3.4. Após verificar em S3 que o valor em AddXi é maior que o número de pontos na memória MD, o controle aguarda no estado S14 pelo fim da última soma iniciada. A Figura 22 apresenta o momento da finalização desta última soma, indicada pelo sinal FimSoma, quando o potencial da última posição de memória é gravado.

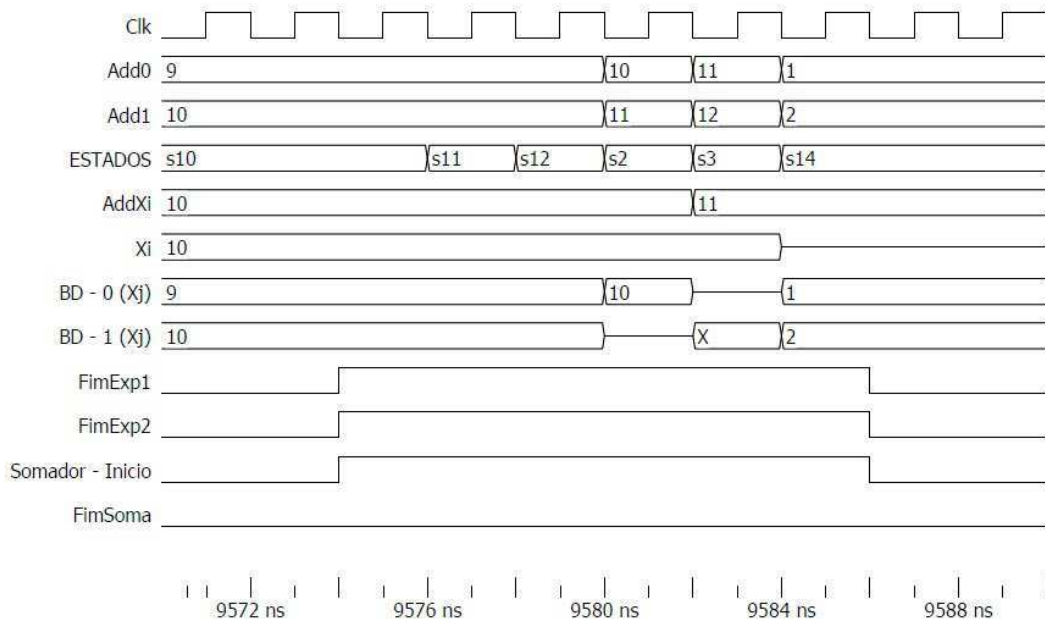


Figura 21: Simulação no momento do reconhecimento do fim dos dados na memória MD

O estado S15 e os próximos preparam os registradores para o início do cálculo de redução. Verifica-se nestes estados, na Figura 22, que o valor xMaxIndice é carregado no registrador AddMD (Add0) para que o barramento BD forneça o valor da memória MD que corresponde ao índice do primeiro centro de agrupamento ( $x_i^*$ ). O valor xMaxIndice é fornecido pela unidade UAS, como representado na Figura 14. No estado seguinte o registrador Xindex é carregado com este valor.

A saída do multiplexador 1 da Figura 10 (representada pelo sinal Xi na Figura 22) passa a fornecer o valor do registrador Xindex da unidade UCAC. Com isto a entrada dos componentes EXP recebem este valor, que será o fator  $x_i^*$  da Equação 9. Os registradores para o numerador (PotMaxN) e denominador (PotMaxD) do potencial máximo, o valor  $P_i^*$  da Equação 9, também são carregados. Com isto o estado S21 inicia os componentes EXP para o cálculo exponencial da Equação 9. O fim deste cálculo é esperado no estado S24.

A Figura 23 apresenta o momento em que o controle reconhece o fim do cálculo expo-

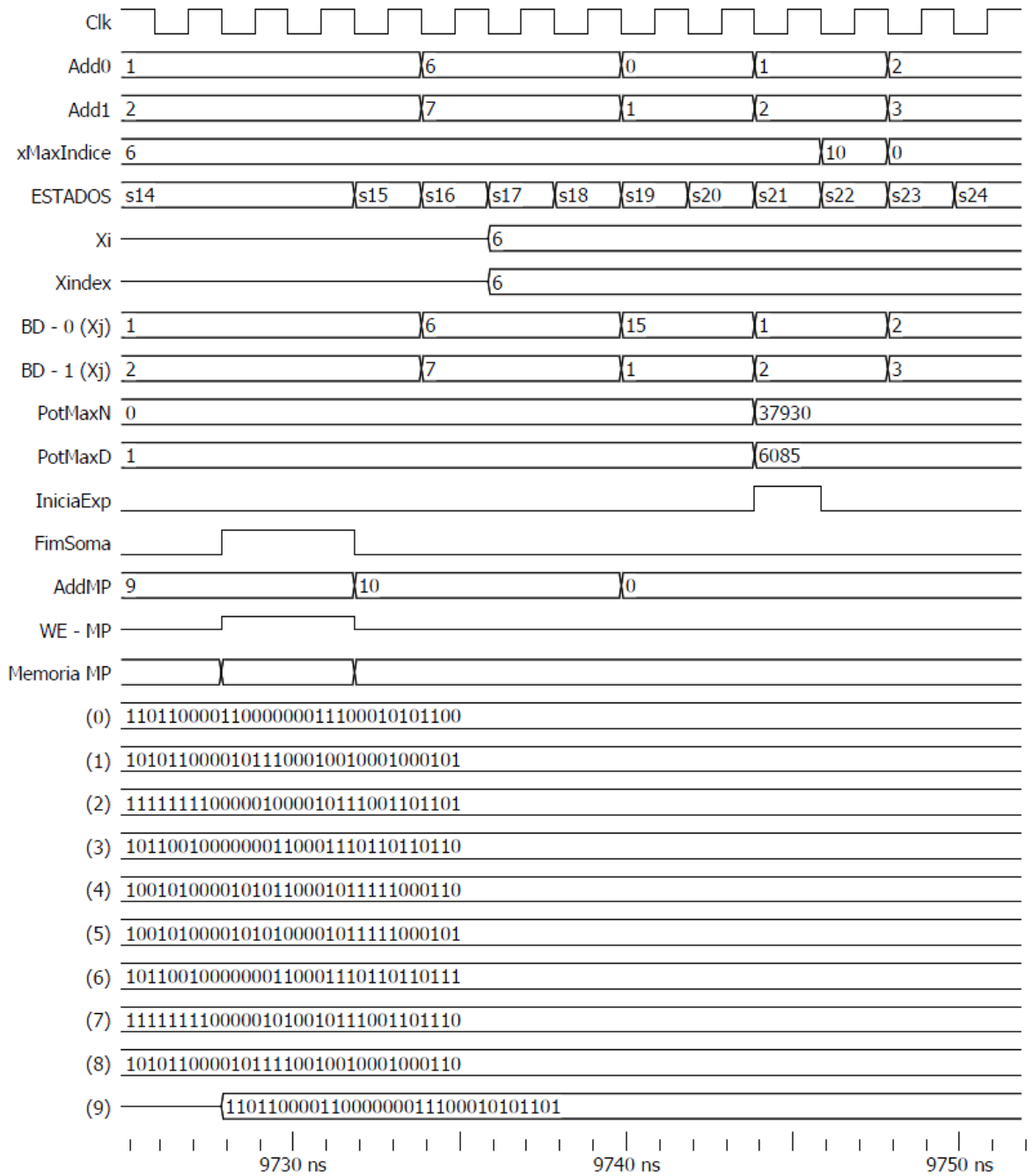


Figura 22: Gravação dos dados e início da redução de potencial

nencial e inicia, com o sinal **Somador-Início**, o restante do cálculo da Equação 9, que inclui uma multiplicação e uma subtração de frações. Este cálculo é realizado no componente **SOMADOR** da unidade **UAS**. O sinal **IniciaPot** em 1 sinaliza à máquina de estados do componente **SOMADOR** que o cálculo deve ser para a Equação 9.

Somente o valor da fração resultado no componente **EXP1** é utilizado. Os detalhes deste

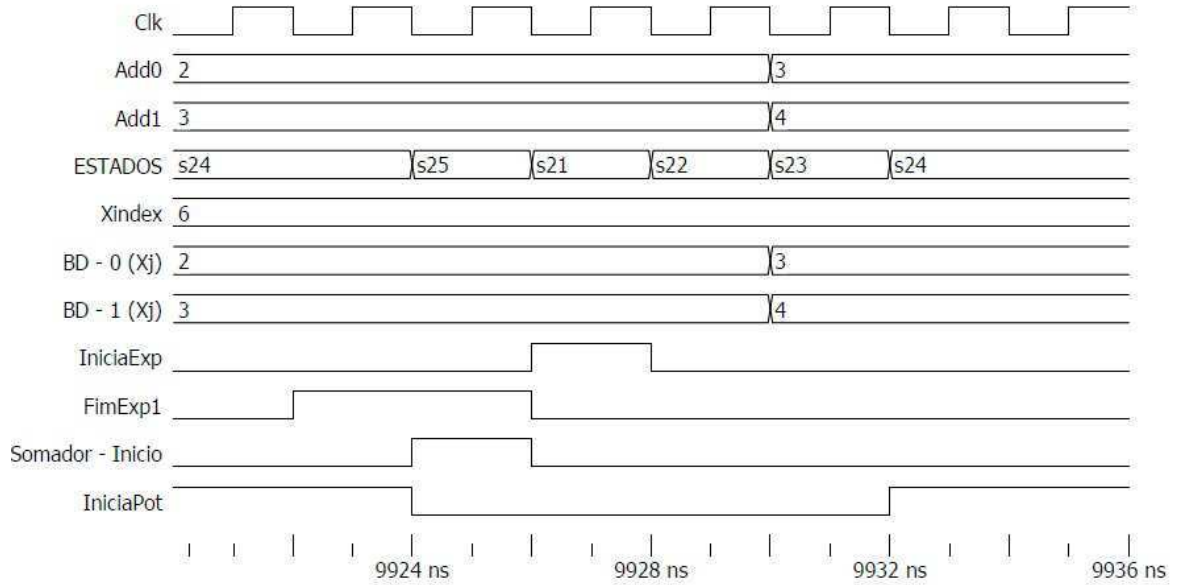


Figura 23: Fim do primeiro cálculo exponencial na redução de potencial

cálculo serão mostrados nas simulações apresentadas no Capítulo 4.

Após este evento o controle pode iniciar um novo cálculo exponencial com um novo valor  $x_i$ , dado por BD-0. A Figura 24 mostra o fim do cálculo iniciado no componente SOMADOR, que habilita a gravação de novo valor potencial na posição 0 da memória MP. Este processo se repete até que todos os valores potenciais tenham sido reduzidos e um novo valor máximo potencial seja calculado.

### 3.6 Considerações Finais do Capítulo

Este Capítulo descreveu a macro-arquitetura proposta em linhas gerais. Foi apresentado em mais detalhes o componente UCAC, responsável pelo controle principal da arquitetura, além de ser responsável por armazenar as amostras para cálculo e os dados de potencial calculados, utilizados para descobrir o centro de agrupamento no processamento do algoritmo de agrupamento subtrativo.

Os sinais de controle entre a UCAC e a UAS, a unidade que realiza os cálculos aritméticos no algoritmo de agrupamento subtrativo, foram apresentados. A representação numérica e a aproximação por polinômios, utilizadas neste projeto, foram apresentadas de forma introdutória, mostrando a importância dessas escolhas para o desenvolvimento da arquitetura proposta. Também foram apresentadas as janelas de simulação das principais etapas realizadas pelo controle para a execução do algoritmo.

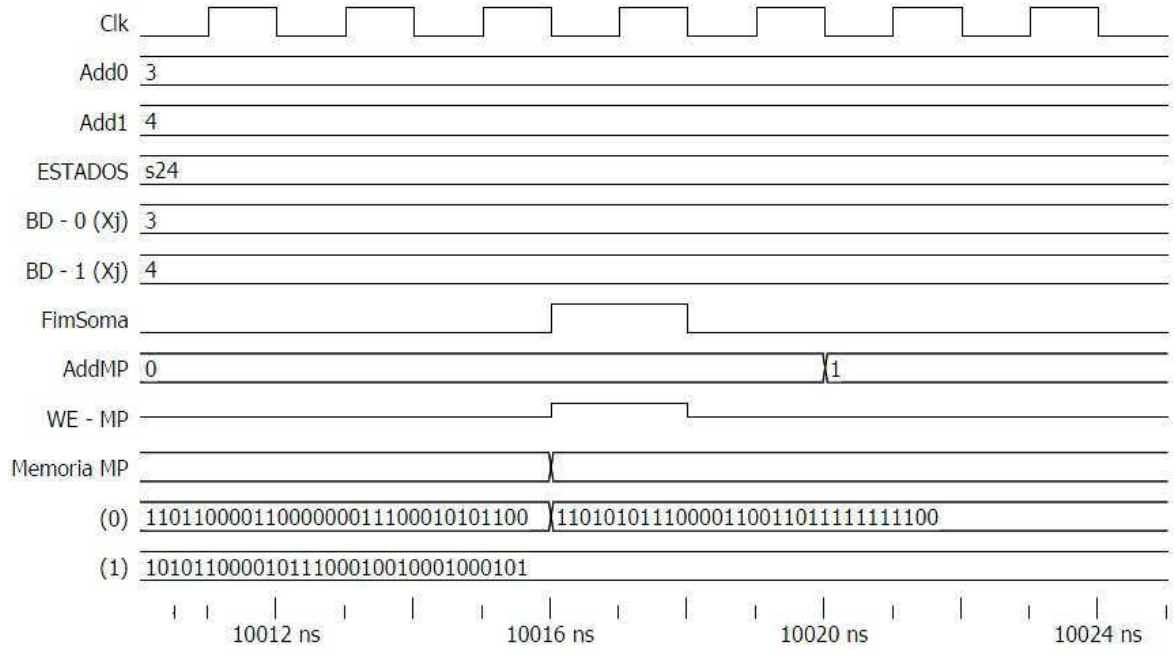


Figura 24: Fim da primeira redução de potencial e gravação do resultado

O Capítulo 4 irá detalhar os componentes da UAS, mostrando a arquitetura e o controle dos componentes para o cálculo do valor exponencial  $e^{-v}$  (EXP), o uso da representação numérica por frações de inteiros e como a arquitetura manipula esses dados.

A arquitetura de *hardware* para realizar o somatório das frações, resultantes dos dois componentes EXP, também será apresentada, bem como o controle para acumular o resultado neste somatório e registrar o potencial máximo calculado, com o índice correspondente da memória dupla MD. Este índice corresponde ao resultado buscado, ou seja, o centro do agrupamento.

## Capítulo 4

# ARQUITETURA DA UNIDADE DE AGRUPAMENTO

NESTE capítulo serão apresentados os detalhes da arquitetura para o cálculo do somatório de valores exponenciais da Equação 8 mostrada no Capítulo 3. Na unidade de *hardware* UAS, descrita sucintamente no Capítulo 3, se processa toda a aritmética para o cálculo desta equação. Esta unidade possui dois componentes EXP, que podem, cada um, calcular um fator do somatório de valores exponenciais em paralelo com o segundo. A UAS possui também, como já visto na Figura 10, um componente para realizar o somatório dos valores calculados pelos dois componentes EXP.

A Seção 4.1 estabelece o modelo para as operações aritméticas realizadas na UAS, as operações com frações de inteiros já mencionadas: somas, subtrações, multiplicações e deslocamentos binários.

A Seção 4.2 mostra os detalhes da arquitetura do componente EXP, que realiza a computação do valor exponencial  $e^{-v}$ , incluindo simulações da operação desta arquitetura. A Seção 4.3 apresenta os detalhes do componente responsável pelo somatório da Equação 8, incluindo também simulações da operação.

A Seção 4.4 apresenta os detalhes para uma implementação escalável da arquitetura, com a simulação desta operação com quatro componentes UAS em paralelo. Na Seção 4.5 são apresentadas as considerações finais do capítulo e um resumo sobre o assunto tratado no Capítulo 5.

### 4.1 Operações Aritméticas

No Capítulo 3 foi visto que o valor da exponencial  $e^{-v}$ , a ser calculado pelo *hardware*, é aproximado por polinômios quadráticos pelo Método dos Mínimos Quadrados (MMQ) (RUGGIERO; LOPES, 1988). Foi visto também que as operações aritméticas executadas pelo *hardware*

baseiam-se em operações com frações de inteiros. Desta forma, um polinômio quadrático pode ser ajustado à exponencial natural  $e^{-v}$ , num intervalo definido de  $v$ , pela Equação 14.

$$e^{-v} = \frac{Na}{Da} \left( \frac{Nv}{Dv} \right)^2 + \frac{Nb}{Db} \left( \frac{Nv}{Dv} \right) + \frac{Nc}{Dc} \quad (14)$$

onde  $v = Nv/Dv$ . Os fatores  $Na/Da$ ,  $Nb/Db$  e  $Nc/Dc$  são os coeficientes para o polinômio de aproximação num intervalo definido de  $Nv/Dv$ .

Verifica-se pela Equação 14 que o *hardware* deve manipular as operação de soma, subtração ou multiplicação com frações, resultando em outra fração, que pode ser representada na notação numerador natural e denominador natural diferente de zero, como pode ser visto nas operações da Equação 11 no Capítulo 3.

Para buscar o melhor desempenho na execução dessas operações pelo *hardware*, foram identificadas as operações que podem ser executadas em paralelo. Essa identificação definiu a arquitetura do *hardware* para o cálculo do valor exponencial, os componentes EXP da unidade UAS. A computação da Equação 14 pode ser realizada pelos passos definidos a seguir:

- Passo 1: Calcula  $A = Nv \times Nv$ ,  $B = Nb \times Nv$ ,  $C = Dv \times Dv$  e  $D = Db \times Dv$ ;
- Passo 2: Calcula  $E = A \times Na$ ,  $F = B \times Dc$ ,  $G = D \times Nc$  e  $H = D \times Dc$ ;
- Passo 3: Calcula  $I = F + G$  e  $J = Da \times C$ ;
- Passo 4: Soma das frações  $E/J + I/H$ . Calcula  $K = E \times H$ ,  $L = J \times I$  e  $Dp = J \times H$
- Passo 5: Calcula  $Np = K + L$ . Resultado em forma de fração:  $Np/Dp$

Observando as operações necessárias para o cálculo do valor exponencial da Equação 14, verifica-se que quatro operações de multiplicação ao mesmo tempo (passo 1 e 2, por exemplo) seriam suficientes para tornar a arquitetura do *hardware* do componente EXP a mais paralela possível para a obtenção do resultado. Esse fato determinou o uso de quatro multiplicadores nesta arquitetura, como será mostrado na Seção 4.2, associado a um circuito somador/subtrator, registradores e multiplexadores.

O resultado do componente EXP, em forma de frações de inteiros, será somado e acumulado em outra unidade da UAS, o SOMADOR. Considerando as duas frações a serem somadas,  $Np1/Dp1$  e  $Np2/Dp2$ , os seguintes passos são necessários:

- Passo 1: Calcula  $A = Np1 \times Dp2$ ,  $B = Np2 \times Dp1$ ,  $D = Dp1 \times Dp2$ ;
- Passo 2: Calcula  $N = A + B$ . Resultado em forma de fração:  $N/D$

Portanto, para a soma de frações são necessários três multiplicadores para realizar em paralelo as operações. Baseado nessa necessidade, o circuito do componente SOMADOR foi desenvolvido com três multiplicadores, como será mostrado na Seção 4.3, associado também a um circuito somador/subtrator, registradores e multiplexadores.

## 4.2 Unidade para o Cálculo Exponencial

Esta seção descreve a unidade de *hardware* EXP que realiza o cálculo da exponencial  $e^{-v}$  utilizando uma aproximação por polinômios e uma representação em forma de frações de inteiros.

A Seção 4.1 apresentou o modelo de cálculo do valor exponencial, o que definiu a montagem desta unidade com os componentes de *hardware* necessários para torná-la a mais paralela possível. A Figura 25 apresenta esta arquitetura.

Na arquitetura existem quatro valores numéricos de  $n$  bits que serão usados para realizar o cálculo exponencial pela Equação 14 de aproximação polinomial, onde  $v = Nv/Dv$  é definido pela Equação 10, apresentada no Capítulo 3. Os valores são  $\alpha$  (registrado internamente na unidade),  $x_j$ ,  $x_i$  e  $x_{max}$ , fornecidos aos componentes EXP, como mostrado na Figura 10 do Capítulo 3. Com esses dados o controle define as operações para a obtenção do valor definido na Equação 10, como será visto mais adiante na descrição da máquina de estados. Com o valor  $Nv/Dv$  calculado conforme a Equação 10, as operações seguintes serão processadas como frações de inteiros, como definido pela Equação 14 e de acordo com os passos descritos na Seção 4.1.

As subseções seguintes descrevem detalhes das operações aritméticas processadas no *hardware*, com explicações da atuação do controlador em cada etapa.

### 4.2.1 Deslocamentos

Na arquitetura descrita na Figura 25 os multiplicadores recebem dados numéricos de  $n$  bits. De forma generalizada esta operação produz dados de  $2n$  bits que são armazenados nos registradores correspondentes (Reg1 a Reg4). Portanto, uma operação aritmética (soma ou multiplicação) entre duas frações,  $Na/Da$  e  $Nb/Db$ , deve passar por um ajuste para se manter a estrutura binária com  $n$  bits no numerador e no denominador. A esse processo chama-se *enquadramento*, que consiste em deslocamentos à direita no registrador que armazena o resultado do numerador, acompanhado de deslocamentos à direita no registrador que armazena o valor do denominador, um bit por vez, até que a fração obtida por esse processo apresente numerador e denominador enquadrados, cada um, em  $n$  bits. A fração gerada após o processo

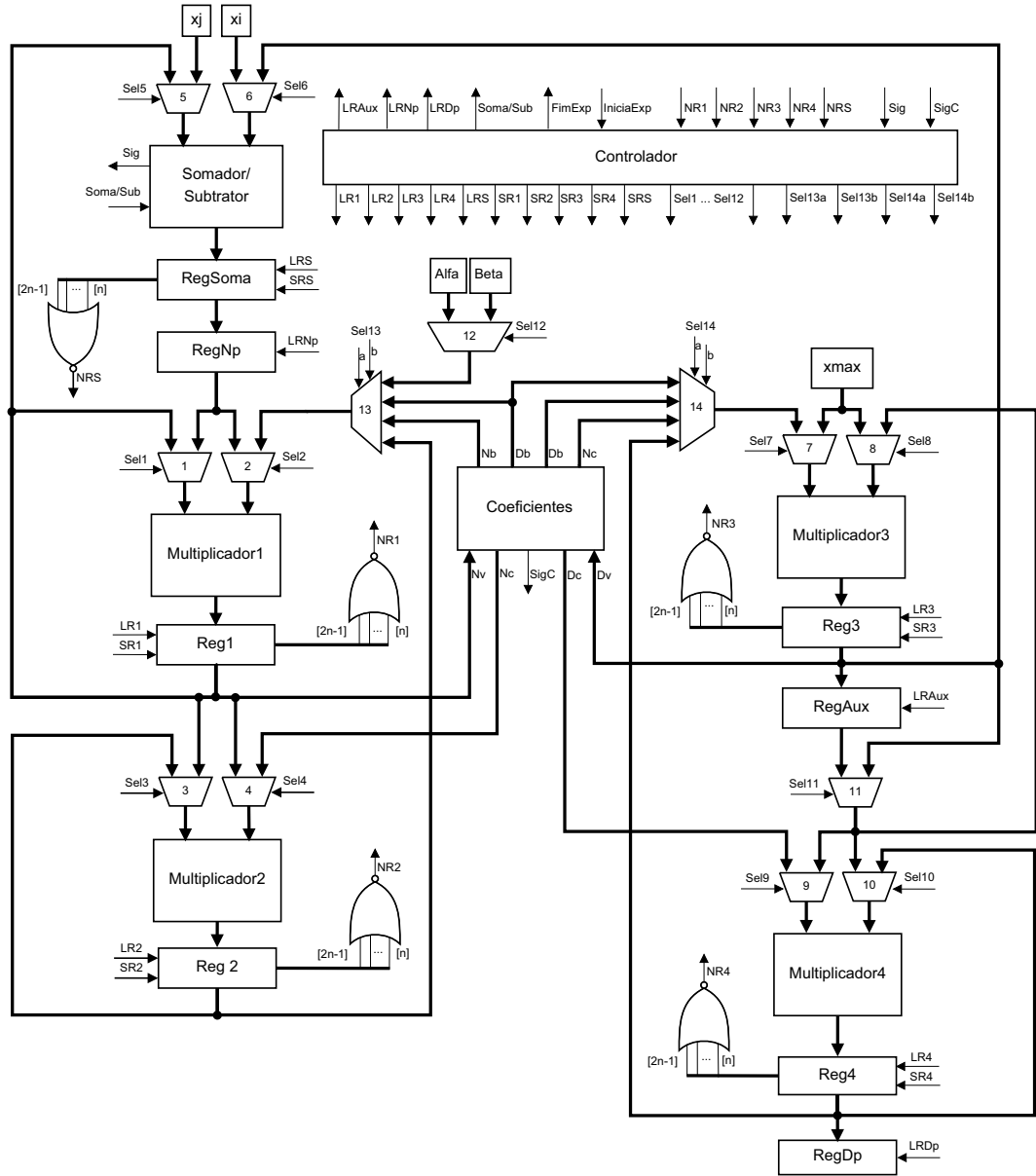


Figura 25: Arquitetura da unidade Exp

de enquadramento é uma aproximação da fração original, visto que em cada deslocamento de 1 bit à direita há uma perda na exatidão da fração, desde que o numerador ou o denominador seja ímpar.

Os registradores Reg1 a Reg4 da arquitetura possuem  $2n$  bits e permitem que o controle realize deslocamentos à direita nesses registradores atuando no sinal de entrada  $SRx$ . O registrador RegSoma também possibilita deslocamentos à direita, através do sinal  $SRS$ , mas possui  $2n + 1$  bits, pois usa a saída de *carry* do somador como bit mais significativo.

Nestes deslocamentos à direita o bit mais significativo do registrador recebe o valor



zero. Cada um desses registradores possui uma porta NOR associada que recebe a metade mais significativa do registrador  $[(2n-1)..n]$ . Assim, somente quando todos os bits na entrada desta porta NOR de  $n$  bits estiverem em zero, a sua saída NRx vai a nível 1.

O sinal de saída dessas portas NOR é usado pelo controle para estabelecer se são necessários mais deslocamentos à direita nos registradores que armazenam o numerador e o denominador durante o processo de enquadramento. Por exemplo, para verificar se o registrador do Multiplicador1 (Reg1 - armazenando o numerador) e o registrador do Multiplicador3 (Reg3 - armazenando o denominador) estão com a estrutura binária correta, o controle recebe os sinais NR1 e NR3 e verifica se os dois estão em nível 1. Caso um deles ou os dois estejam em nível zero, é indicador de que o numerador ou o denominador, ou ambos, estão com valor numérico que excede o número de bits  $n$  do padrão. Com isso, deslocamentos à direita nos dois registradores são efetuados pelo controle até que os sinais NR1 e NR3 estejam com valor 1.

A situação em que o denominador é muito menor que o numerador, tal como a fração  $Q/P$ , onde  $Q \ll P$ , no exemplo da Equação 15, deve ser tratada para que não ocorra uma divisão por zero ao final do processo de enquadramento.

$$\frac{Q}{P} = \frac{2175426}{7} \rightarrow \frac{1087713}{3} \rightarrow \frac{543856}{1} \rightarrow \frac{271928}{0} \rightarrow \frac{135964}{0} \rightarrow \frac{67982}{0} \rightarrow \frac{33991}{0} \quad (15)$$

Considerando o número de bits  $n$  do padrão igual a 16, a última fração deslocada (obtida após o sexto deslocamento) não está enquadrada na estrutura padrão corretamente, porque o denominador se tornou igual a zero. Assim, quando o denominador atinge o valor 1 e caso o numerador não seja representável em  $n$  bits, então irá ocorrer uma divisão por zero no próximo deslocamento. Neste caso, os deslocamentos devem ser interrompidos pelo controle. A fração que melhor se adequa, neste caso, refere-se ao valor máximo representado na Equação 16, enquadrado na estrutura binária com  $n$  bits,  $n$  igual a 16.

$$\frac{1111111111111111}{0000000000000001} = 2^{16} - 1 = 65535 \quad (16)$$

#### 4.2.2 Coeficientes para aproximação por polinômios

A unidade de *hardware* COEFICIENTES fornece os seis fatores das frações  $Na/Da$ ,  $Nb/Db$  e  $Nc/Dc$ , usados no polinômio de aproximação definido pela Equação 14. Para isso esta unidade de *hardware* deve descobrir em que intervalo está o fator  $Nv/Dv$ , onde  $Nv/Dv$  é dada pela Equação 10.

Cada intervalo definirá um conjunto de fatores, como visto no Capítulo 3, estabelecendo um polinômio de segunda ordem. O *hardware* COEFICIENTES precisa, portanto, descobrir em

que intervalo se encontra o valor  $Nv/Dv$  dentre os seguintes:  $Nv/Dv \in [0, 1[$ ,  $Nv/Dv \in [1, 2[$ ,  $Nv/Dv \in [2, 4[$ ,  $Nv/Dv \in [4, 8[$  e  $Nv/Dv \in [8, +\infty[$ . A fração  $Nv/Dv$  é calculada no início do processamento da função exponencial e fornecida às entradas  $Nv$  e  $Dv$  de COEFICIENTES.

A Figura 26 mostra a unidade COEFICIENTES. A unidade possui um registrador de deslocamento à direita (RegNv), um subtrator, uma porta NOR e um pequeno controle baseado em máquina de estados. Dependendo do estado da máquina, um conjunto de fatores é fornecido à saída de COEFICIENTES. Portanto a máquina possui cinco estados possíveis.

Os intervalos para aproximar a função exponencial por polinômios foram escolhidos de forma que fornecessem, além de uma boa aproximação, uma simplificação para a descoberta dos coeficientes pelo *hardware*. Dessa forma, a colocação dos limites dos intervalos em potências de 2 foi conveniente.

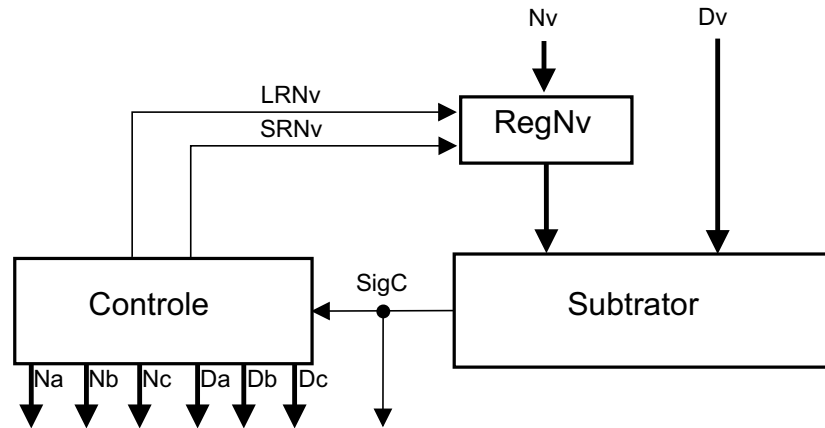
Com a unidade COEFICIENTES é possível subtrair o denominador  $Dv$  do numerador  $Nv$  para compará-los. O valor de  $Nv$  chega até o subtrator pelo registrador com deslocamento à direita RegNv. Se  $Nv$  é menor que  $Dv$  o valor para o qual irá se calcular a exponencial é menor que 1, ou seja, está no intervalo  $[0, 1[$ . Isso se descobre pelo sinal no resultado da operação no SUBTRATOR da Figura 26. Assim, o conjunto de fatores correspondente a esse intervalo é disponibilizado nas saídas  $Na$ ,  $Da$ ,  $Nb$ ,  $Db$ ,  $Nc$  e  $Dc$ .

Caso o sinal da operação de subtração ( $Nv - Dv$ ) dê positivo, o valor no numerador  $Nv$ , registrado em RegNv, é deslocado uma vez para direita, com o bit mais significativo recebendo zero. Isso representa uma divisão por dois no valor do numerador  $Nv$ , ou por consequência, uma divisão por dois no valor da fração  $Nv/Dv$ . Quando se desloca à direita o valor  $Nv$ , o controle também atualiza o conjunto de fatores para agora fornecer os correspondentes ao intervalo  $[1, 2[$ . Uma nova subtração ( $Nv - Dv$ ) é realizada, sendo que o valor de  $Nv$ , fornecido por RegNv, foi dividido por dois em relação ao valor original. Caso o sinal da subtração indique que o novo valor de  $Nv$  é menor que  $Dv$ , significa que o valor original da fração  $Nv/Dv$  está no intervalo  $[1, 2[$ .

Esse processo é repetido para se descobrir se o intervalo está entre  $[2, 4[$ ,  $[4, 8[$  ou se é maior ou igual a oito, caso após três deslocamentos à direita o valor de  $Nv$  continue maior ou igual a  $Dv$ .

### 4.2.3 Controlador

Os sinais de entrada e saída do controlador do componente EXP são mostrados na Figura 25. Este controlador tem como função fornecer os sinais necessários para realizar os passos

Figura 26: Unidade de *hardware* COEFICIENTES

definidos na Seção 4.1 para a computação da Equação 14. A função desses sinais de controle está descrita a seguir. Todos estes são sinais de 1 bit.

1. **IniciaExp**: Sinal recebido do controlador principal. Inicia, quando igual a 1, a execução do cálculo do valor exponencial no componente EXP, retirando a máquina de estados deste controle do estado inicial. Deste modo a computação do valor exponencial  $e^{-v}$ , onde  $v$  é dado pela Equação 10, é iniciada com os valores de  $x_j$ ,  $x_i$  e  $x_{max}$  disponibilizados ao componente EXP.
2. **NR1 a NR4 e NRS**: Estes sinais são usados pelo controle para verificar a necessidade de deslocamentos à direita no processo de enquadramento. Por exemplo, se o controle está verificando a necessidade de enquadramento nos registradores 1 e 3 (numerador e denominador nessa ocasião), somente quando os dois sinais NR1 e NR3 estiverem em 1 os sinais para deslocá-los à direita é retirado.
3. **Sig**: Fornecido pelo somador/subtrator. Sinaliza, quando em 1, que o sinal da operação de subtração foi negativo. Se zero, sinaliza que foi positivo.
4. **SigC**: Fornecido pelo subtrator da unidade de *hardware* COEFICIENTES. Da mesma forma, sinaliza, quando em 1, que o sinal da operação de subtração foi negativo. Se zero, sinaliza que foi positivo.
5. **LR1 a LR4 e LRS**: Estes sinais são fornecidos pelo controle para carregar, na transição de um para zero, valores nos registradores correspondentes de cada multiplicador (1 a 4) ou no registrador do somador/subtrator.

6. **SR1 a SR4 e SRS**: Estes sinais, junto com o sinal de carregamento correspondente, realiza o deslocamento à direita nos registradores Reg1 a Reg4 e no registrador do somador/subtrator (RegSoma). Portanto, o controlador usa estes sinais nos diversos processos de enquadramento.
7. **Selx**: O controlador atua com estes sinais nos multiplexadores da arquitetura, permitindo que o valor necessário em cada estágio do controle chegue aos multiplicadores e ao somador/subtrator para o cálculo do polinômio.
8. **LRAux**: Sinal do controle para carregar o registrador auxiliar, necessário em um dos estágios do cálculo do polinômio aproximador.
9. **LRNp e LRDp**: São os sinais do controlador para o registrador dos valores do resultado final, numerador no registrador RegNp e denominador no registrador RegDp.
10. **Soma/Sub**: Sinal do controlador para selecionar a operação de subtração, quando em 1, no somador/subtrator. Quando em zero, seleciona a função de soma.

A máquina de estados deste controlador é mostrada na Figura 27. A descrição de cada estado é mostrada a seguir e está baseada nos passos apresentados na Seção 4.1.

- **S0**: Inicia o sistema; prepara a primeira operação, a subtração ( $x_j - x_i$ ). Para isso, seleciona as entradas correspondentes dos multiplexadores 5 e 6, além da operação de subtração (**Soma/Sub** = 1).  
Se sinal **IniciaExp** é apresentado, vai para o estado S1;
- **S1**: Prepara o carregamento do registrador RegSoma com o valor absoluto da subtração no próximo estado (S2);
- **S2**: Prepara as multiplicações  $\text{RegSoma} \times \text{RegSoma}$  e  $x_{\max} \times x_{\max}$ ; Seleciona a entrada RegSoma dos multiplexadores 1 e 2 e a entrada  $x_{\max}$  do multiplexador 7. Assim Reg1 terá o resultado do numerador  $(x_j - x_i)^2$  e Reg3 o resultado do denominador  $(x_{\max})^2$ ;
- **S3**: Carrega os registradores Reg1 e Reg3;
- **S4**: Enquadramento dos registradores da fração Reg1/Reg3. Se os sinais **NR1** e **NR3** são iguais a 1, não realiza deslocamentos e vai para o S6. Se não, vai para o estado S5;
- **S5**: Desloca Reg1 e Reg3 um bit à direita e retorna a S4;

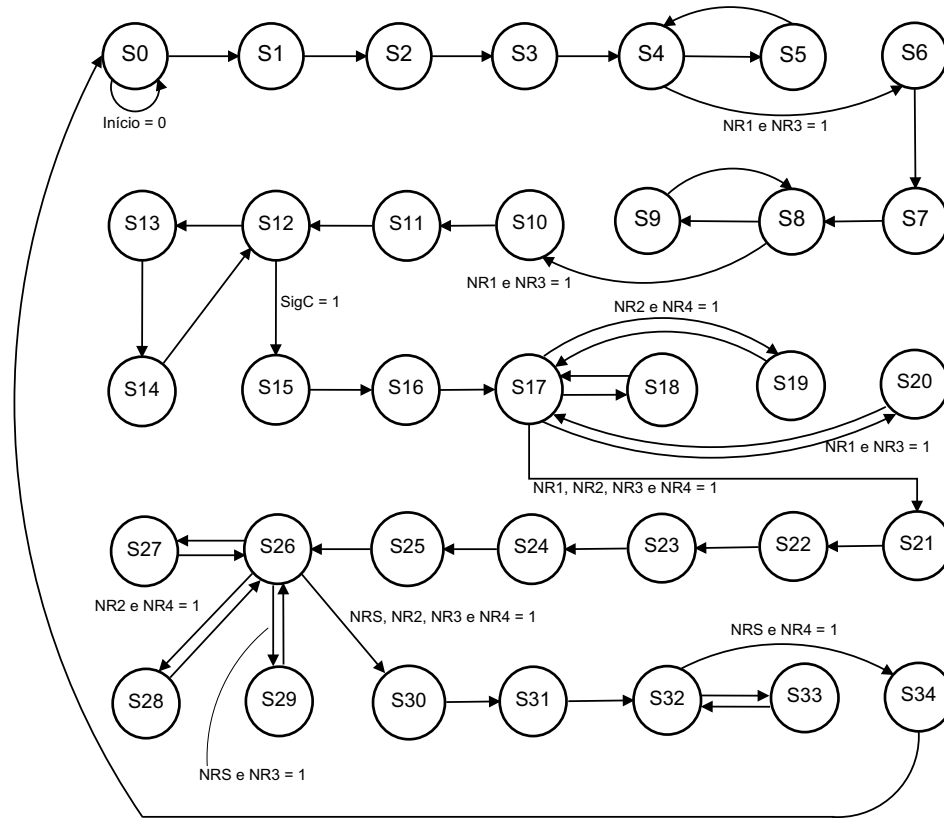


Figura 27: Máquina de estados do controlador do componente EXP

- S6: Prepara a multiplicação do numerador (Reg1) por  $\alpha$ ; Seleciona a entrada Reg1 do multiplexador 1 e  $\alpha$  do multiplexador 2;
- S7: Carrega o registrador Reg1 com o valor da multiplicação;
- S8: Enquadramento dos registradores da fração Reg1/Reg3. Se os sinais NR1 e NR3 são iguais a 1, não realiza deslocamentos e vai para o S10. Se não, vai para o estado S9;
- S9: Desloca Reg1 e Reg3 um bit à direita e retorna a S8;;
- S10: O valor da fração  $N_v/D_v$  foi calculado. Inicia a busca dos coeficientes Na, Da, Nb, Db, Nc e Dc;
- S11: Carrega o registrador da unidade COEFICIENTES com o valor do numerador  $N_v$ ;
- S12: Se o sinal no resultado do subtrator da unidade COEFICIENTES é 1,  $N_v$  é menor que  $D_v$  e a fração é menor 1. Vai para S15 e sai do enquadramento; do contrário vai para S13;

- S13: Deslocamento à direita no valor  $N_v$  do registrador da unidade COEFICIENTES para dividi-lo por 2;
- S14: Muda para próximo estado do controle em COEFICIENTES para selecionar fatores correspondentes e retorna a S12.
- S15: Com os fatores do polinômio descobertos, realiza quatro multiplicações simultâneas. Reg1 tem o valor  $N_v$  e Reg3 o valor  $D_v$ . As entradas dos multiplexadores são selecionadas para que as seguintes multiplicações ocorram:  $\text{Reg2} = \text{Reg1} \times \text{Reg1}$ ,  $\text{Reg1} = \text{Reg1} \times N_b$ ,  $\text{Reg3} = \text{Reg3} \times D_b$  e  $\text{Reg4} = \text{Reg3} \times \text{Reg3}$ ;
- S16: Carrega os registradores Reg1, Reg2, Reg3 e Reg4;
- S17: Enquadramento dos registradores das frações Reg1/Reg3 e Reg2/Reg4. Se o sinal NR1 ou NR3 é igual a zero e NR2 ou NR4 é igual a zero, vai para S18;  
Do contrário se o sinal NR1 ou NR3 é igual a zero, e NR2 e NR4 são iguais a um, vai para S19;  
Do contrário se o sinal NR1 e NR3 são iguais a um, e NR2 ou NR4 é igual a zero, vai para S20;  
Do contrário se o sinal NR1 e NR3 são iguais a um, e NR2 e NR4 também são iguais a um, vai para S21;
- S18: Desloca Reg1, Reg2, Reg3 e Reg4 um bit à direita e retorna a S17;
- S19: Desloca Reg1 e Reg3 um bit à direita e retorna a S17;
- S20: Desloca Reg2 e Reg4 um bit à direita e retorna a S17;
- S21: Fim do enquadramento. Carrega RegAux com o valor de Reg3 ( $D_v \times D_b$ ) para ser usado posteriormente. Realiza mais quatro multiplicações simultâneas. Reg1 tem o valor  $N_v \times N_b$ , Reg2 tem o valor  $N_v \times N_v$ , Reg3 tem o valor  $D_v \times D_b$  e Reg4 tem o valor  $D_v \times D_v$ . As entradas dos multiplexadores são selecionadas para que as seguintes multiplicações ocorram:  $\text{Reg2} = \text{Reg2} \times N_a$ ,  $\text{Reg1} = \text{Reg1} \times D_c$ ,  $\text{Reg3} = \text{Reg3} \times N_c$  e  $\text{Reg4} = \text{Reg4} \times D_a$ ;
- S22: Carrega os registradores Reg1, Reg2, Reg3 e Reg4;
- S23: O coeficiente  $N_b$  da Equação 13 de aproximação, dada no Capítulo 3, é negativo em todos os intervalos. Logo neste estado é feita a operação de subtração  $\text{Reg3} -$

Reg1, onde Reg3 possui o valor  $D_v \times D_b \times N_c$  e Reg1 possui o valor  $N_v \times N_b \times D_c$ . Para isso seleciona-se as entradas correspondentes dos multiplexadores para a operação de subtração. O resultado desta operação estará em RegSoma.

- S24: Carrega o registrador RegSoma com o resultado da operação de subtração. Guarda o sinal da operação para saber se o resultado foi negativo ou positivo.

Realiza a multiplicação do registrador RegAux, que possui o valor  $D_v \times D_b$ , por  $D_c$ . Seleciona os multiplexadores para que o multiplicador 3 realize essa operação.

- S25: Carrega o registrador Reg3 com o valor da operação. Logo  $\text{Reg3} = D_v \times D_b \times D_c$ ;
- S26: Enquadramento dos registradores das frações  $\text{RegSoma}/\text{Reg3}$  e  $\text{Reg2}/\text{Reg4}$ . Se o sinal NRS ou NR3 é igual a zero e NR2 ou NR4 é igual a zero, vai para S27;

Do contrário se o sinal NRS ou NR3 é igual a zero, e NR2 e NR4 são iguais a um, vai para S28;

Do contrário se o sinal NRS e NR3 são iguais a um, e NR2 ou NR4 é igual a zero, vai para S29;

Do contrário se o sinal NRS e NR3 são iguais a um, e NR2 e NR4 também são iguais a um, vai para S30;

- S27: Desloca RegSoma, Reg2, Reg3 e Reg4 um bit à direita e retorna a S26;
- S28: Desloca RegSoma e Reg3 um bit à direita e retorna a S26;
- S29: Desloca Reg2 e Reg4 um bit à direita e retorna a S26;
- S30: Fim do enquadramento. A operação final será somar essas duas frações que guardam os resultados parciais:  $\text{RegSoma}/\text{Reg3} + \text{Reg2}/\text{Reg4}$ . Para isso, três multiplicações simultâneas são feitas:  $\text{Reg3} = \text{Reg2} \times \text{Reg3}$ ,  $\text{Reg1} = \text{Reg4} \times \text{RegSoma}$  e  $\text{Reg4} = \text{Reg3} \times \text{Reg4}$ .
- S31: Carrega os registradores Reg1, Reg3 e Reg4 com os resultados das operações. Seleciona o sinal para a próxima operação entre Reg3 e Reg1.  
Se o sinal guardado no estado S24 foi negativo, seleciona a subtração ( $\text{Soma}/\text{Sub} = 1$ ).  
Do contrário, seleciona a soma ( $\text{Soma}/\text{Sub} = 0$ ).
- S32: Carrega o resultado da soma ou subtração em RegSoma. A operação foi selecionada em S31.

Enquadramento dos registradores das frações RegSoma/Reg4, que guardam o resultado final do cálculo do polinômio de aproximação da função exponencial. Se os sinais NRS e NR4 são iguais a 1, não realiza deslocamentos e vai para o S34. Se não, vai para o estado S33;

- S33: Desloca RegSoma e Reg4 um bit à direita e retorna a S32;
- S34: Fim. Gera o sinal FimExp para indicar o fim do processo. Carrega os registradores RegNp e RegDp e retorna a S0.

#### 4.2.4 Simulações

Esta Seção apresenta as janelas de simulação da unidade EXP. As principais etapas são apresentadas para mostrar a execução do algoritmo, cujos passos estão detalhados na Seção 4.2.3.

A Figura 28 mostra o início da operação neste componente, quando o sinal **IniciaExp** é recebido do controle principal em UCAC. Neste momento a UCAC já enviou os sinais para registrar os valores  $x_i$  e  $x_{max}$  (10 e 15 neste exemplo) e aguarda no estado S10 o fim do cálculo nos componentes EXP, como visto na Seção 3.5. Os valores  $x_j$  para o componente EXP1 (igual a 10) e EXP2 (igual a 2) já estão disponibilizados no barramento BD. Esses correspondem aos endereços 1 e 2 da memória MD.

Verifica-se que os dois componentes EXP recebem o sinal **IniciaExp** e começam a executar o cálculo em paralelo. No início uma subtração é efetuada ( $x_j - x_i$ ) com o módulo do resultado aparecendo em RegSoma. No estado S3 são carregados os registradores Reg1, com o valor ao quadrado de RegSoma e Reg3 com o valor ao quadrado de  $x_{max}$ .

O estado S4 verifica a necessidade de enquadramento dos bits na fração Reg1/Reg3. O estado S6 prepara a multiplicação do numerador (Reg1) por  $\alpha$ , com o resultado carregado no próprio Reg1 em S7. O estado S8 verifica novamente a necessidade de enquadramento.

O estado S10 inicia o processo de busca dos coeficientes. A Figura 29 mostra este processo nos dois componentes EXP. Para obter os coeficientes corretos, o algoritmo busca o intervalo em que se encontra a fração Reg1/Reg3 através da verificação do sinal da subtração e deslocamentos à direita, como explicado na Seção 4.2.2. Em S15 os coeficientes corretos estão descobertos e disponíveis. A Figura 29 mostra apenas os coeficientes  $Na$  e  $Da$ , mas todos são buscados e disponibilizados ao mesmo tempo.

Como em EXP1 a fração Reg1/Reg3 é igual a zero, o coeficiente é encontrado mais rápido e o processamento do algoritmo segue, como descrito na Seção 4.2.3. Em EXP2 a fração está entre 4 e 8, levando mais ciclos para ser encontrada.



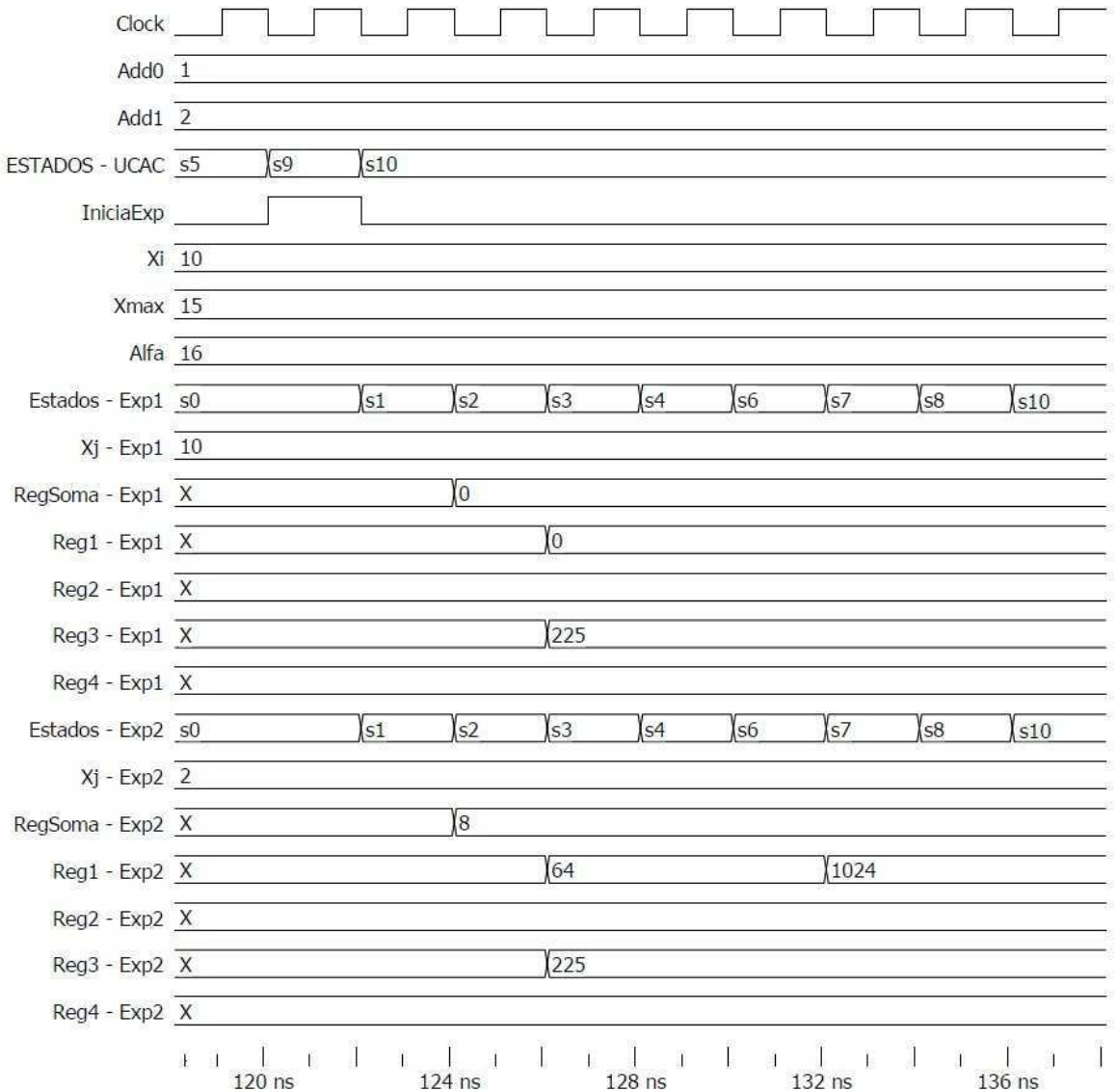


Figura 28: Início da operação nos componentes EXP

Na Figura 29 observa-se que quando o componente EXP2 encontrou os coeficientes corretos, em S15, o componente EXP1 já havia realizado várias operações (até o estado S25) para o cálculo do polinômio quadrático dado pela Equação 14. Essas diferenças no tempo de processamento, que ocorrem também no enquadramento das frações, se refletem no tempo total de processamento de cada componente EXP da UAS, fazendo com que o esse término possa ocorrer em intervalo de tempo distinto.

A Figura 30 apresenta o final do processamento nos componentes EXP, com o resultado nos registradores RegSoma (numerador) e Reg4 (denominador). O cálculo é finalizado antes em EXP1 (sinal **FimExp1**=1) com o retorno ao estado inicial. Neste momento EXP2 está realizando

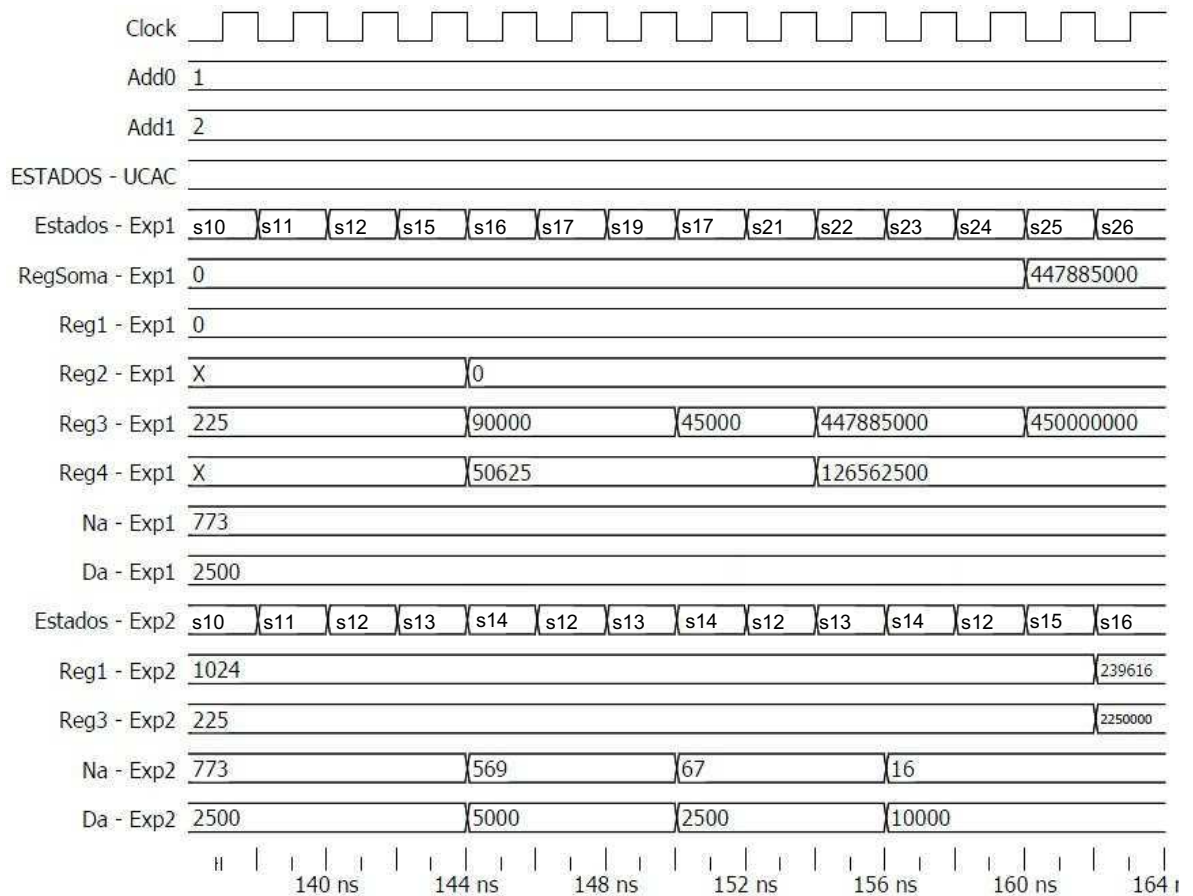


Figura 29: Simulação na busca dos coeficientes nos componentes EXP

o enquadramento em RegSoma e Reg4. Com os dois sinais **FimExp** em 1, o estado do controle principal em UCAC vai para S11, como visto no Capítulo 3 nas Seções 3.4 e 3.5.

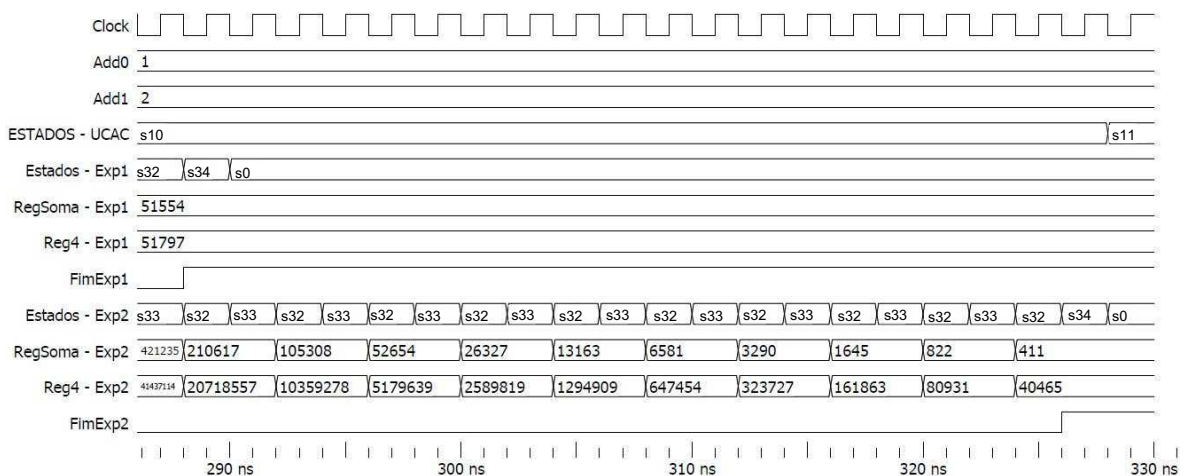


Figura 30: Fim do processamento nos componentes EXP

As simulações da unidade para o somatório da UAS, que serão mostradas na Seção 4.3.2, começarão a partir deste ponto, no recebimento dos dois sinais `FimExp=1`.

## 4.3 Unidade para o Somatório

Nesta seção será descrito o modelo proposto para o componente de *hardware* SOMADOR. Em seguida, simulações para validar seu correto funcionamento são apresentadas e explicadas.

### 4.3.1 Modelo proposto

A Figura 31 apresenta a arquitetura do componente SOMADOR. Este componente realiza a soma das frações calculadas nos componentes EXP e disponibilizadas nas saídas Np (numerador) e Dp (denominador) de cada um. Cada fração gerada em cada componente EXP, como visto na Seção 4.2, representa um fator  $e^{-v}$ , onde o valor  $v$  é igual a  $Nv/Dv$ , dado pela Equação 10. Cada fator deste é usado no somatório para o cálculo do potencial de cada amostra do conjunto de amostras, dado pela Equação 8.

O componente SOMADOR possui, portanto, quatro entradas numéricas: duas denominadas Np1 e Np2, representando os numeradores dos fatores calculados em cada um dos dois componentes EXP, e duas denominadas Dp1 e Dp2, representando os denominadores de cada fator. As duas entrada de sinais `FimExp1` e `FimExp2` indicam o término do processo nos componentes EXP. Com os dois sinais indicando o fim do processo, o componente SOMADOR inicia a soma dos fatores automaticamente.

O componente deve realizar também, além da soma dos dois fatores recebidos, a acumulação das somas parciais até que todos os fatores do somatório tenham sido calculados pelos componentes EXP. Os registradores RegAuxN e RegAuxD têm a função de guardar as somas parciais do somatório, como será visto na descrição da máquina de estados do controlador da unidade.

Ao final do processo de somatório, o cálculo do valor potencial de uma amostra do conjunto de dados estará concluído. Neste momento o componente SOMADOR disponibiliza o valor, em forma de fração, ao barramento de potenciais BP para que seja gravado na memória de potenciais MP, como visto na Figura 10 do Capítulo 3.

Outra função do componente SOMADOR é, ao final de cada somatório, comparar o potencial calculado com o anterior para guardar o potencial máximo nos registradores RegMaxN e RegMaxD, além do índice máximo no registrador RegIndice. Este último é o resultado do

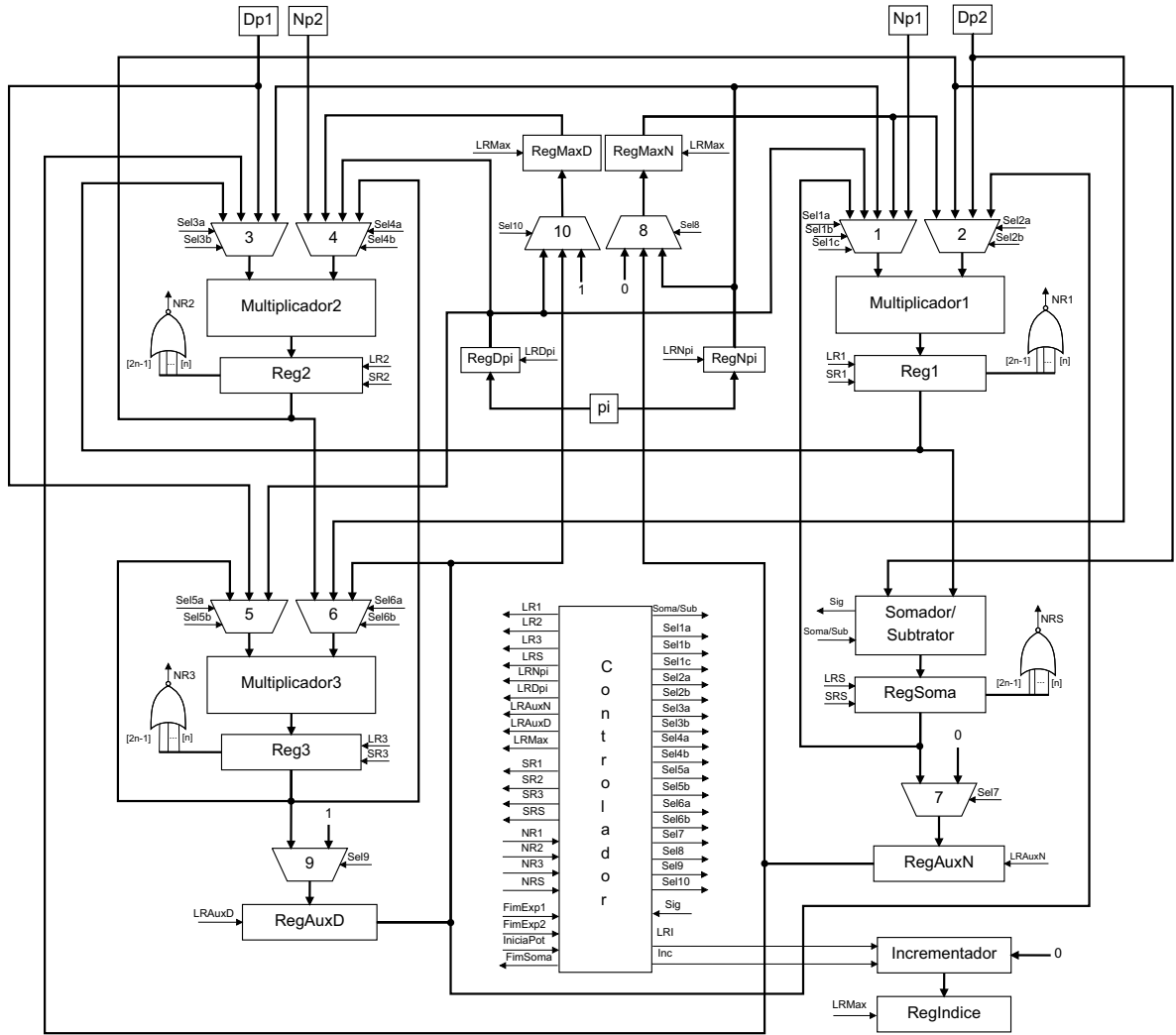


Figura 31: Arquitetura interna do componente SOMADOR

processo de agrupamento subtrativo, pois guarda a posição da memória MD cuja a amostra tem o potencial máximo, ou seja, o centro do agrupamento.

Para comparar o potencial calculado com anteriores é preciso realizar uma comparação de frações para descobrir a de maior valor. A fração que guarda o valor potencial recém calculado está em RegAuxN/RegAuxD, e a fração que guarda o potencial máximo até o momento está RegMaxN/RegMaxD. Para descobrir qual a maior fração, deve-se realizar a operação para tornar os denominadores iguais. Para isso, multiplica-se os membros da primeira fração pelo denominador da segunda e, de forma análoga, os membros da segunda fração pelo denominador da primeira. Com os denominadores iguais ( $\text{RegAuxD} \times \text{RegMaxD}$ ), o numerador da fração 1 fica com o valor  $\text{RegAuxN} \times \text{RegMaxD}$  e o numerador da fração 2 fica com o valor  $\text{RegMaxN} \times \text{RegAuxD}$ . A fração cujo numerador possuir o maior valor após essa operação é

a maior fração. Com isso, o controlador pode descobrir a maior fração apenas fazendo duas multiplicações e uma subtração, como será visto nos detalhes da descrição de sua máquina de estados.

A Seção 4.1 definiu os passos e os componentes necessários para tornar esta arquitetura a mais paralela possível. Como trabalha com operações aritméticas de frações de inteiros, assim como o componente EXP, esta unidade tem semelhanças com esta última. Os registradores associados aos multiplicadores 1, 2 e 3 e ao somador/subtrator da unidade também são de deslocamento à direita. O processo de enquadramento também será necessário nesta arquitetura, semelhantemente ao descrito na Seção 4.2.1.

Além de realizar o somatório, esta unidade também será responsável por realizar a redução de potenciais descrita na Equação 9 do Capítulo 3. O controlador principal da arquitetura dará o sinal para que este processo de redução se inicie. O cálculo exponencial da Equação 9 é feito em um dos componentes EXP da UAS, sob comando do controlador principal em UCAC, como visto no Capítulo 3. O componente EXP gerará os valores  $Np1$  e  $Dp1$ , que será usado pelo componente SOMADOR para completar o cálculo da Equação 9.

Os sinais de entrada e saída do controlador do componente SOMADOR são mostrados na Figura 31. A função desses sinais de controle está descrita a seguir. Todos estes são sinais de 1 bit.

1. **FimExp1** e **FimExp2**: Sinais recebidos dos dois componentes EXP. Quando os dois vão a 1, inicia a execução da soma das duas frações de entrada:  $Np1/Dp1$  e  $Np2/Dp2$ ;
2. **NR1** a **NR3** e **NRS**: Estes sinais, provenientes das portas NOR associadas aos registradores Reg1, Reg2, Reg3 e RegSoma, são usados pelo controle para verificar a necessidade de deslocamentos à direita no processo de enquadramento.
3. **Sig**: Fornecido pelo somador/subtrator. Sinaliza, quando em 1, que o sinal da operação de subtração foi negativo. Se zero, sinaliza que foi positivo.
4. **IniciaPot**: Fornecido pelo controlador principal da arquitetura em UCAC, este sinal inicia a redução de potenciais, após o final do processo de cálculo dos potenciais de todos os pontos.
5. **LR1** a **LR3** e **LRS**: Estes sinais são fornecidos pelo controle da unidade para carregar, na transição de um para zero, valores nos registradores correspondentes de cada multiplicador (1 a 3) ou no registrador do somador/subtrator.

6. **SR1 a SR3 e SRS**: Estes sinais, junto com o sinal de carregamento correspondente, realiza o deslocamento à direita do registradores Reg1 a Reg3 e do registrador do somador/subtrator. Portanto, o controlador usa estes sinais nos diversos processos de enquadramento.
7. **Selx**: O controlador atua, com estes sinais, nos multiplexadores da arquitetura, permitindo que o valor necessário em cada estágio do controle chegue aos multiplicadores e ao somador/subtrator para o cálculo do somatório de frações.
8. **LRAuxN e LRAuxP**: Sinais do controle para carregar o registradores auxiliares (numerador e denominador), necessário para a guarda das somas parciais do somatório.
9. **LRMax**: Sinal do controle para carregar os registradores do potencial máximo (RegMaxN - numerador e RegMaxD - denominador) e também para carregar o índice do potencial máximo no registrador RegIndice.
10. **LRDpi**: Sinal do controle para carregar os registradores do potencial guardado na memória MP (RegNpi - numerador e RegDpi - denominador) no processo de redução de potenciais.
11. **Inc**: Este sinal é enviado para incrementar o registrador que guarda o índice atual da memória (INCREMENTADOR).
12. **LRI**: Este sinal é enviado para carregar com zero o registrador INCREMENTADOR.
13. **Soma/Sub**: Sinal do controlador para selecionar a operação de subtração, quando em 1, no somador/subtrator. Quando em zero, seleciona a função de soma.
14. **FimSoma**: Sinaliza o fim do processo de somatório.

A máquina de estados deste controlador é mostrada na Figura 32. A descrição de cada estado é mostrada a seguir e está baseada nos passos apresentados na Seção 4.1.

- **S0**: Inicia o sistema; carrega com zero os registradores RegAuxN e RegMaxN e com 1 os registradores RegAuxD e RegMaxD. Isso é feito para tornar as frações iguais a zero. Carrega com zero o registrador RegIndice.  
Prepara as primeiras operações simultâneas, as 3 multiplicações para somar as frações  $Np1/Dp1$  e  $Np2/Dp2$ :  $Reg1 = Np1 \times Dp2$ ,  $Reg2 = Np2 \times Dp1$  e  $Reg3 = Dp1 \times Dp2$ ;  
Se os sinais **FimExp1** e **FimExp2** vão a 1, vai para o estado S2 para iniciar o somatório;  
Se **IniciaPot** vai a 1, muda para S19 para iniciar a redução de potenciais;

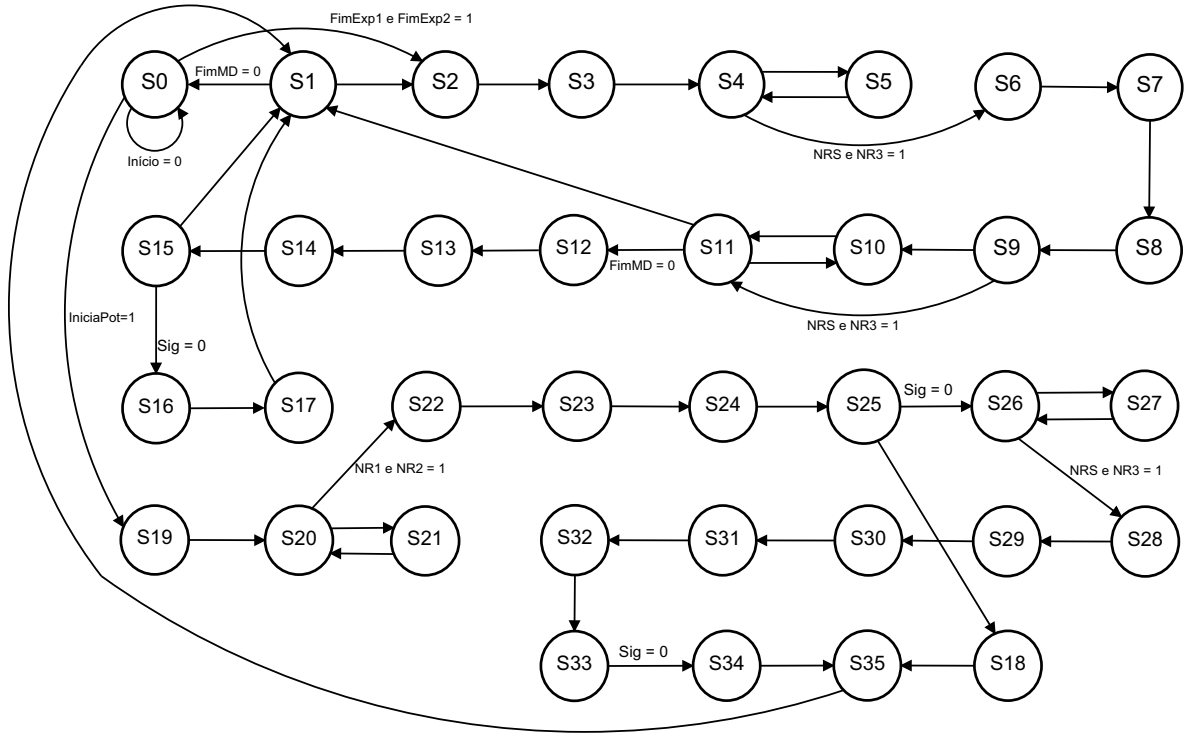


Figura 32: Máquina de estados do controlador do componente SOMADOR

- S1: Prepara as operações simultâneas, as 3 multiplicações para somar as frações  $Np1/Dp1$  e  $Np2/Dp2$ :  $Reg1 = Np1 \times Dp2$ ,  $Reg2 = Np2 \times Dp1$  e  $Reg3 = Dp1 \times Dp2$ ;  
Se os sinais **FimExp1** e **FimExp2** vão a 1, vai para o estado S2 para iniciar o somatório;  
Se **IniciaPot** vai a 1, muda para S19 para iniciar a redução de potenciais;  
Se todos os dados  $x_j$  da memória MD foram somados, vai para S0;
- S2: Carrega os registradores **Reg1**, **Reg2** e **Reg3**;
- S3: Prepara a soma de **Reg1** e **Reg2**;
- S4: Carrega o registradores **RegSoma**;  
Enquadramento dos registradores da fração **RegSoma/Reg3**. Se os sinais **NRS** e **NR3** são iguais a 1, não realiza deslocamentos e vai para o S6. Se não, vai para o estado S5;
- S5: Desloca **RegSoma** e **Reg3** um bit à direita e retorna a S4;
- S6: Prepara as multiplicações para somar as frações **RegSoma/Reg3** e **RegAuxN/RegAuxD**, onde a última guarda o somatório acumulado até o momento.  $Reg1 = RegSoma \times RegAuxD$ ,  $Reg2 = Reg3 \times RegAuxN$  e  $Reg3 = Reg3 \times RegAuxD$ ;
- S7: Carrega os registradores **Reg1**, **Reg2** e **Reg3**; Prepara a soma de **Reg1** e **Reg2**;

- S8: Carrega o registrador RegSoma;
- S9: Enquadramento dos registradores da fração RegSoma/Reg3. Se os sinais NRS e NR3 são iguais a 1, não realiza deslocamentos e vai para o S11. Se não, vai para o estado S10;
- S10: Desloca RegSoma e Reg3 um bit à direita e retorna a S10;
- S11: Carrega os registradores RegAuxN e RegAuxD, resultado parcial do somatório;  
Se todos os dados  $x_j$  da memória MD foram somados para calcular o potencial do ponto  $x_i$  (sinal FimSoma = 1), vai para S12;  
Do contrário, volta para S1 para somar mais dois fatores  $x_j$ .
- S12: O valor potencial é gravado na memória MP.  
Inicia a verificação se a fração do potencial calculado (RegAuxN/RegAuxD) é maior que o potencial máximo encontrado até aqui (RegMaxN/RegMaxD). Prepara as multiplicações, definindo as entradas dos multiplexadores para que as seguintes operações ocorram:  $\text{Reg2} = \text{RegAuxN} \times \text{RegMaxD}$  e  $\text{Reg1} = \text{RegMaxN} \times \text{RegAuxD}$ .
- S13: Carrega os registradores Reg1 e Reg2;
- S14: Prepara a subtração  $\text{Reg1} - \text{Reg2}$
- S15: Carrega o registrador RegSoma;  
Se o resultado da subtração é positivo, vai para S16; do contrário vai para S1;
- S16: Prepara o carregamento do novo valor em RegMaxN e RegMaxD a partir dos valores de RegAuxN e RegAuxP respectivamente, e o incremento do registrador RegIndice, o índice do potencial máximo, no próximo estado;
- S17: Retorna a S1 para recomençar o somatório para cálculo do potencial da próxima amostra.
- S18: Prepara para gravar o valor potencial na memória MP com valor zero. Vai a S35.
- S19: Inicia a redução de potenciais das amostras para a descoberta do segundo centro de agrupamento. Carrega com zero o registrador RegAuxN e com 1 o registrador RegAuxD. Isso é feito para tornar a fração igual a zero e recomençar a busca de potencial máximo. Carrega com zero o registrador INCREMENTADOR.  
Seleciona as entradas dos multiplexadores para as seguintes operações, definidas na Equação 9:  $\text{Reg1} = \text{RegMaxN} \times \text{Np1}$  e  $\text{Reg2} = \text{RegMaxD} \times \text{Dp1}$ .



- S20: Carrega os registradores Reg1 e Reg2;  
Enquadramento dos registradores da fração Reg1/Reg2. Se os sinais NR1 e NR2 são iguais a 1, não realiza deslocamentos e vai para o S22. Se não, vai para o estado S21;
- S21: Prepara o deslocamento de Reg1 e Reg2 um bit à direita e retorna a S20;
- S22: O valor do potencial a ser reduzido está no barramento BP, disponibilizado pelo controle principal. Este é registrado em RegNpi e RegDpi. Seleciona as entradas dos multiplexadores para as operações de multiplicação na subtração de frações definida na Equação 9:  $\text{Reg1} = \text{Reg2} \times \text{RegNpi}$ ,  $\text{Reg2} = \text{Reg1} \times \text{RegDpi}$  e  $\text{Reg3} = \text{Reg2} \times \text{RegDpi}$ .
- S23: Carrega os registradores Reg1, Reg2 e Reg3;
- S24: Prepara a Subtração de  $\text{Reg1} - \text{Reg2}$ ;
- S25: Carrega o registrador RegSoma; se o resultado da subtração é positivo, vai a S26 para enquadramento.  
Do contrário vai a S18 para gravar zero na memória.
- S26: Enquadramento dos registradores da fração RegSoma/Reg3. Se os sinais NRS e NR3 são iguais a 1, não realiza deslocamentos e vai para o S28. Se não, vai para o estado S27;
- S27: Desloca RegSoma e Reg3 um bit à direita e retorna a S26;
- S28: Seleciona as entradas dos multiplexadores para os registradores de saída terem os valores RegSoma e Reg3.
- S29: Carrega os registradores RegAuxN e RegAuxD com os valores RegSoma e Reg3.
- S30: Inicia a verificação se a fração do potencial calculado ( $\text{RegAuxN}/\text{RegAuxD}$ ) é maior que o potencial máximo encontrado até aqui ( $\text{RegMaxN}/\text{RegMaxD}$ ). Prepara as multiplicações, definindo as entradas dos multiplexadores para que as seguintes operações ocorram:  $\text{Reg2} = \text{RegAuxN} \times \text{RegMaxD}$  e  $\text{Reg1} = \text{RegMaxN} \times \text{RegAuxD}$ .
- S31: Carrega os registradores Reg1 e Reg2;
- S32: Prepara a Subtração de  $\text{Reg1} - \text{Reg2}$ ;
- S33: Carrega o registrador RegSoma; Se o resultado da subtração é positivo, vai para S34;  
Do contrário, vai para S35;

- S34: Prepara o carregamento do novo valor em RegMaxN e RegMaxD a partir dos valores de RegAuxN e RegAuxD respectivamente, e o incremento do registrador RegIndice, o índice do potencial máximo, no próximo estado (S35);
- S35: O sinal FimSoma é posto em 1 para gravar o valor potencial na memória MP. Retorna a S1 para recomençar o cálculo de redução do potencial da próxima amostra.

### 4.3.2 Simulações

Esta Seção apresenta as janelas de simulação da unidade SOMADOR. As principais etapas são apresentadas para mostrar a execução do algoritmo, cujos passos estão detalhados na Seção 4.3.1.

A Figura 33 mostra o início da execução do processo de soma das duas frações calculadas nos componentes EXP (Np1/Dp1 e Np2/Dp2). Estas frações estão nas entradas de mesmo nome da Figura 31.

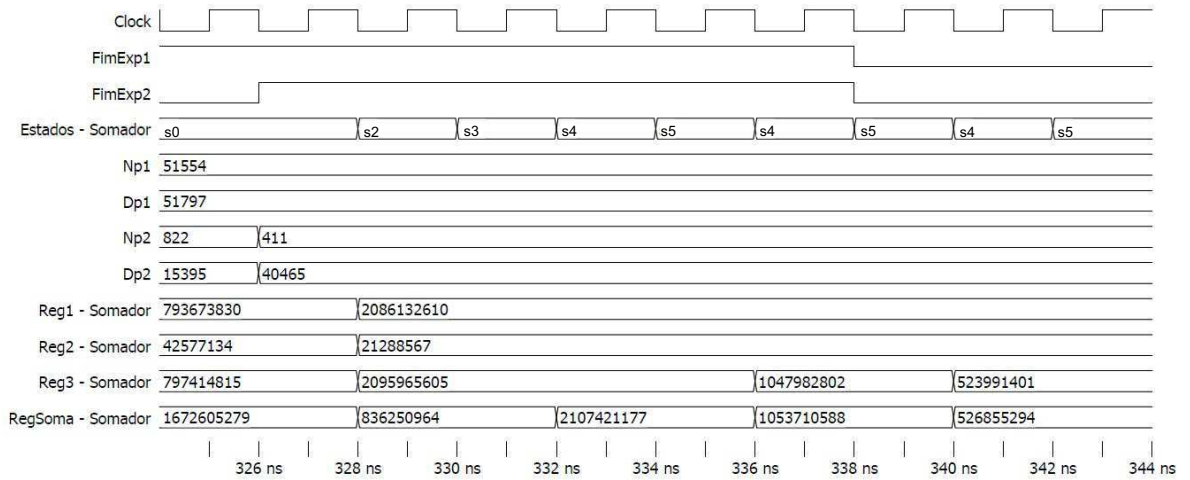


Figura 33: Início da operação no componente Somador

Com os dois sinais **FimExp** em 1, o controle do componente SOMADOR passa ao estado S2 e inicia a soma das frações, já carregando os registradores Reg1, Reg2 e Reg3 com os valores das multiplicações, como visto na descrição dos estados da Seção 4.3.1. Em S5 a fração resultado da soma das frações de entrada (Np1/Dp1 e Np2/Dp2) já está disponível em RegSoma/Reg3, iniciando-se o processo de enquadramento desses registradores.

A Figura 34 mostra o final desse enquadramento. Nos estados seguintes o valor obtido é somado com o valor dos registradores que acumulam as somas parciais, também em forma de fração (RegAuxN/RegAuxD). Neste caso está fração ainda está com valor 0/1.

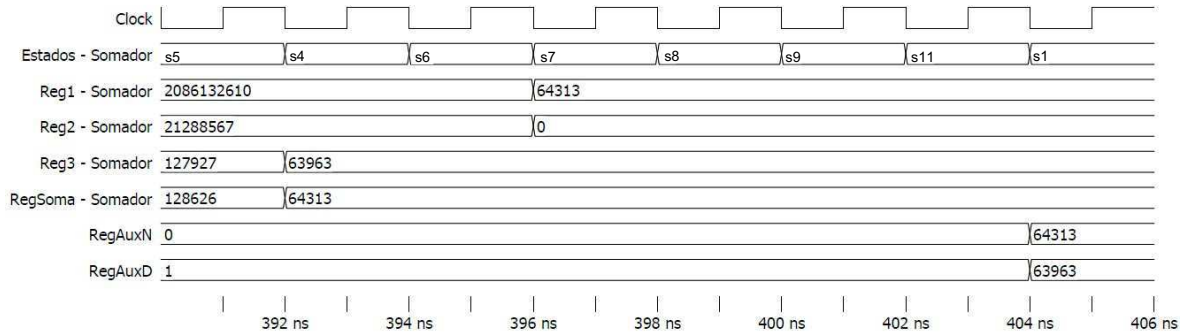


Figura 34: Final da primeira soma no componente Somador

O resultado parcial do somatório é passado para RegAuxN/RegAuxD no estado S1, que espera pelo início de nova soma parcial.

A Figura 35 mostra o final do somatório, indicado pelo sinal **FimSoma**. Isto ocorre depois que todos os fatores  $e^{-v}$  da Equação 8, onde o valor  $v$  é igual a  $Nv/Dv$ , dado pela Equação 10. O valor da fração em RegAuxN/RegAuxD é o potencial da primeira amostra  $x_i$  da memória MD.

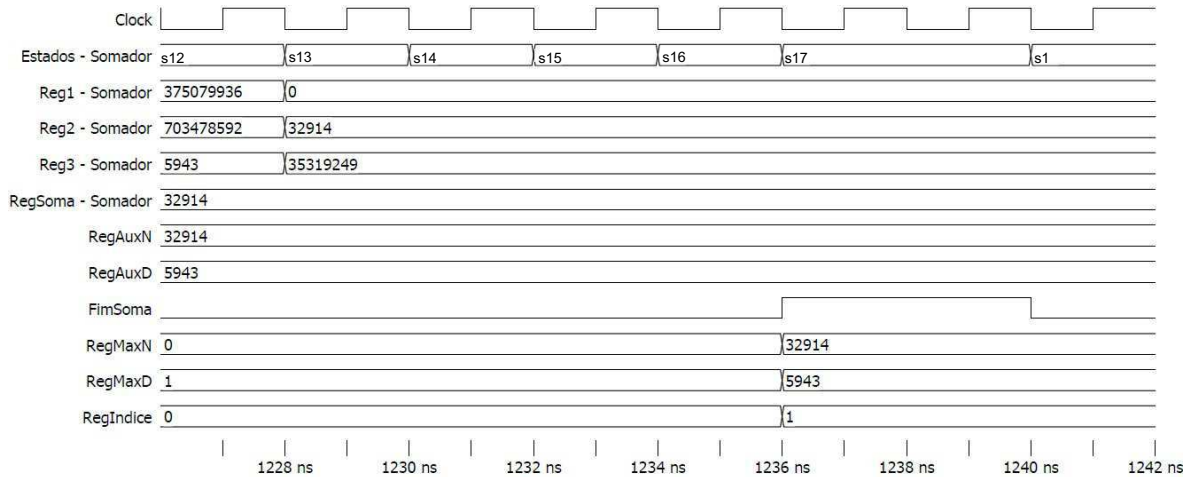


Figura 35: Final do cálculo do primeiro potencial no componente somador

O resultado em RegAuxN/RegAuxD é disponibilizado no barramento BP para ser gravado na memória de potenciais MP. Este valor também é comparado com valor de potencial máximo, registrado em RegMaxN/RegMaxD, que neste momento ainda é inicial (0/1). Como o valor potencial da amostra atual (RegAuxN/RegAuxD) é maior que o valor potencial máximo registrado até agora (RegMaxN/RegMaxD), o novo valor máximo é passado para os

registradores RegMaxN/RegMaxD em S17. O registrador do índice máximo, em RegIndice, também é atualizado com o índice atual.

Esse processo se repete até que todos as amostras tenham o seu potencial calculado. Ao final, o potencial máximo calculado e o índice (posição) na memória MD desta amostra estão armazenados, o que revela o primeiro centro de agrupamento descoberto. A Figura 36 mostra esse momento, em que o último somatório termina e os valores em RegMaxN/RegMaxD são passados a registradores auxiliares para serem usados durante a redução de potencial.

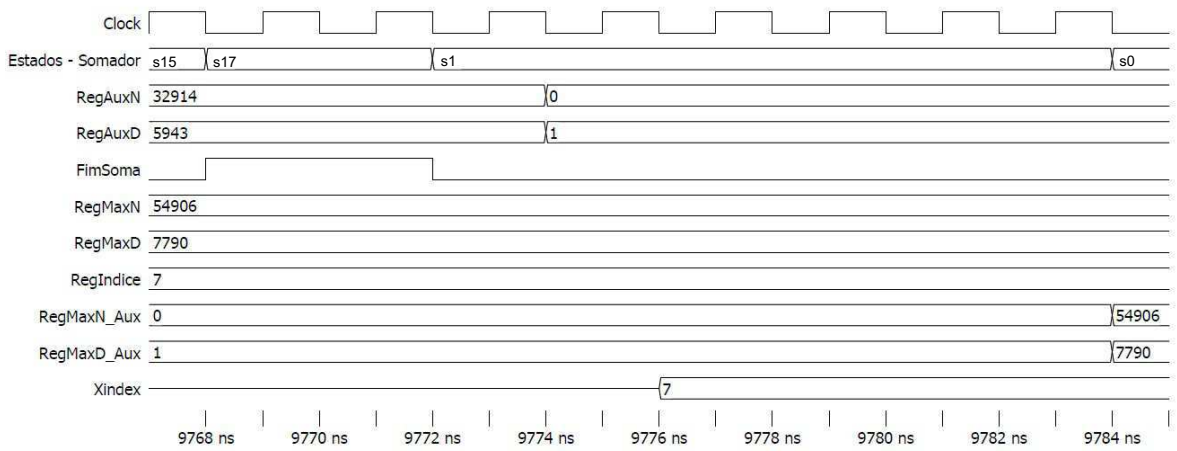


Figura 36: Final do cálculo do último potencial no componente Somador

O valor do registrador RegIndice é o valor passado para a UCAC através de xMaxIndice (Figura 14). Com isto o registrador Xindex da UCAC é carregado, sob comando do controle principal, com o valor da amostra de índice máximo, que é o primeiro centro de agrupamento. Este valor também é usado no processo de redução, sendo o fator  $x^*$  da Equação 9.

A redução do valor potencial de cada ponto do conjunto de amostras começa com o cálculo do valor exponencial da Equação 9. Com o valor exponencial calculado por EXP1, o controle em UCAC inicia o restante do cálculo da Equação 9 enviando o sinal **IniciaPot** ao componente SOMADOR. Este cálculo envolve uma multiplicação de frações, seguida de uma subtração de frações.

A Figura 37 mostra o início desse cálculo. Verifica-se que os registradores de potencial máximo e índice foram reiniciados com zero para este processo.

O valor exponencial calculado ( $Np1/Dp1$ ) é fornecido às entradas de mesmo nome do componente SOMADOR. Este valor é multiplicado pelo valor máximo potencial, guardado nos registradores auxiliares RegMaxNAux/RegMaxDAux, com o resultado (numerador e denomi-

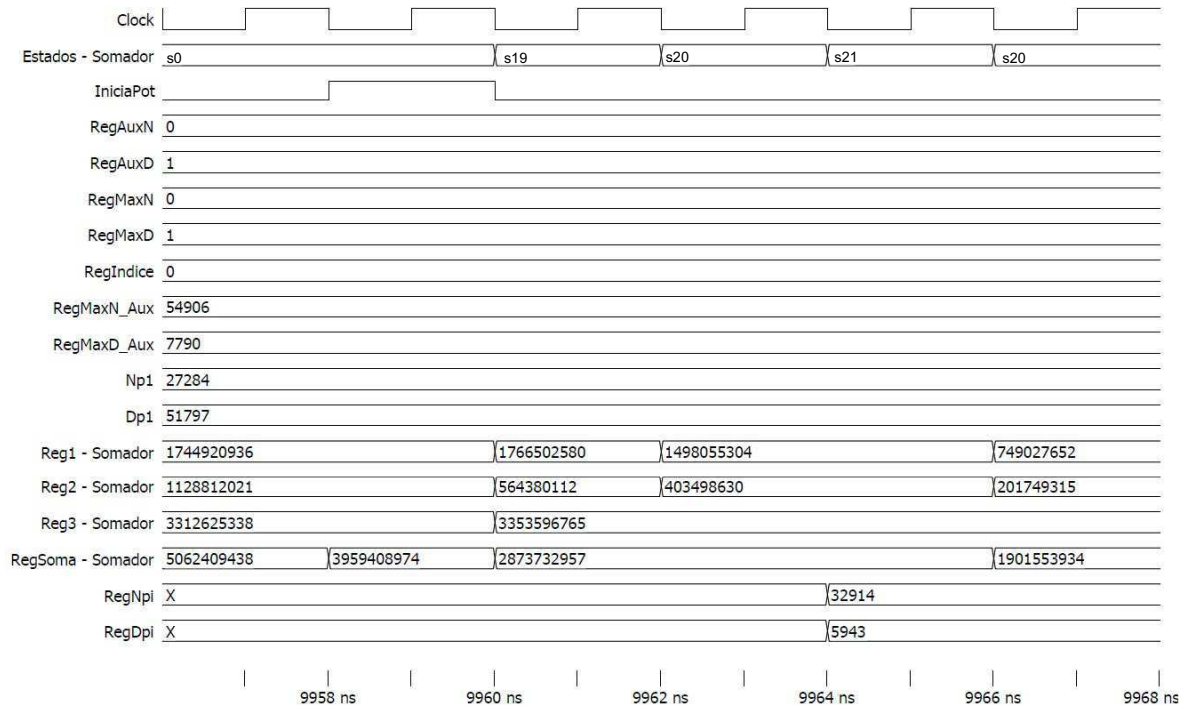


Figura 37: Início da redução de potencial no componente somador

nador) sendo carregado em Reg1 e Reg2, respectivamente, seguindo-se mais um processo de enquadramento nestes registradores.

Pode-se observar que o valor RegAuxN/RegAuxD que aparece na Figura 35, o potencial calculado para a primeira amostra, reaparece em RegNpi e RegDpi. Estes valores foram gravados na memória MP e agora são recuperados para a subtração de frações que ocorrerá em seguida para completar o cálculo da Equação 9 desta primeira amostra. Após essa subtração e de novo enquadramento dos registradores, o cálculo de redução de potencial da primeira amostra é concluído, com o sinal FimSoma em 1 e um novo valor máximo potencial sendo registrado, juntamente com o seu índice. O processo é repetido até que todos os potenciais sejam reduzidos e um segundo centro de agrupamento seja encontrado.

## 4.4 Escalabilidade da Arquitetura

Uma característica importante da arquitetura proposta neste trabalho é possibilitar que mais unidades de agrupamento subtrativo (UAS) sejam acrescentadas para operar em paralelo. Nesta seção, será apresentado o projeto escalável do *hardware* proposto e as simulações para validar seu correto funcionamento.

### 4.4.1 Modelo escalável

Cada componente UAS calcula, em paralelo com as outras unidades, o potencial de um ponto  $i$ , o valor  $P_i$  da Equação 8 apresentada no Capítulo 3. Para isso cada módulo (UAS) em paralelo deve receber e registrar um valor  $x_i$  com o qual trabalhará durante o cálculo do potencial de um ponto.

Para que não seja necessário um aumento no número de sinais de controle fornecidos pelo componente UCAC a medida que novos componentes UAS sejam acrescentados, o próprio componente UAS deve enviar e receber alguns sinais de controle em comunicação com o componente UAS subsequente. A Figura 38 mostra o acréscimo de componentes UAS em paralelo, com os sinais de controle entre os mesmos.

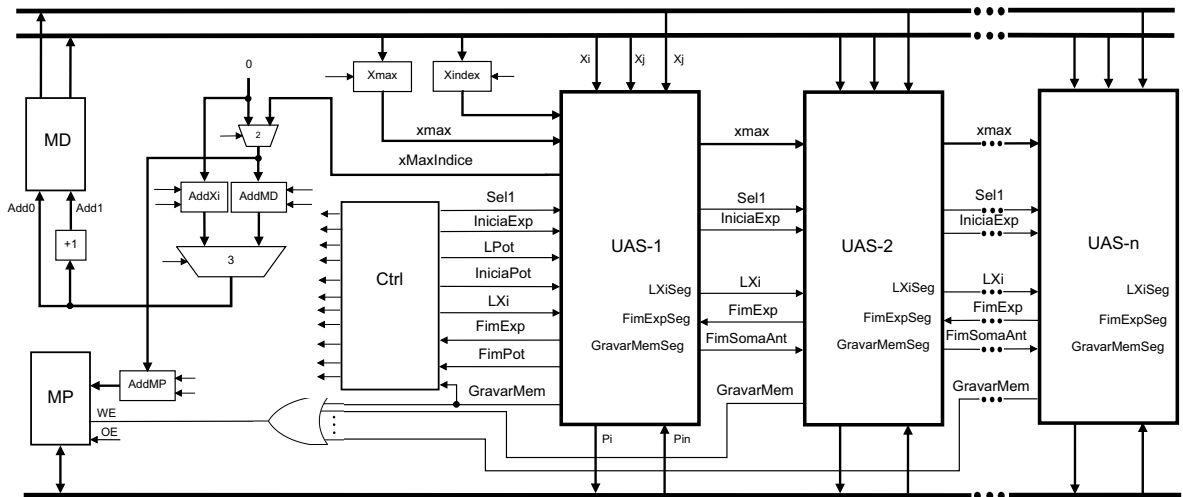


Figura 38: Escalabilidade da arquitetura

O valor  $x_i$  é carregado no registrador  $X_i$  a partir do barramento duplo BD, como mostrado na Figura 10. Como estes valores estão em posições diferentes da memória, este registrador do valor  $x_i$  tem de ser feito em espaço de tempo diferente. Isso porque não é possível aumentar o número de portas da memória a medida que o número de componentes UAS é aumentado.

O sinal  $LXiSeg$  é o sinal para carregar o registrador  $X_i$  no UAS seguinte, ou subsequente, da arquitetura. O UAS possui um pequeno controle (mostrado na Figura 10) que habilita o sinal  $LXiSeg$  alguns ciclos após receber o sinal  $LXi$ , tempo necessário para que o componente UCAC disponibilize o valor  $x_i$  seguinte da memória no barramento BD. Assim, cada componente UAS acrescentado receberá o seu sinal para carregar o valor  $x_i$  no registrador  $X_i$  no tempo correto.

A Figura 39 apresenta uma ilustração no tempo do processo de cálculo da arquitetura com  $n$  unidades UAS em paralelo. Partindo da esquerda para a direita, o processo se inicia com a distribuição dos valores  $x_i$  armazenados nos endereços da memória MD. Os endereços são dados por  $i * n + 1$  até  $i * n + n$ , onde  $i$  começa com zero na primeira distribuição e  $n$  representa o número de UAS.

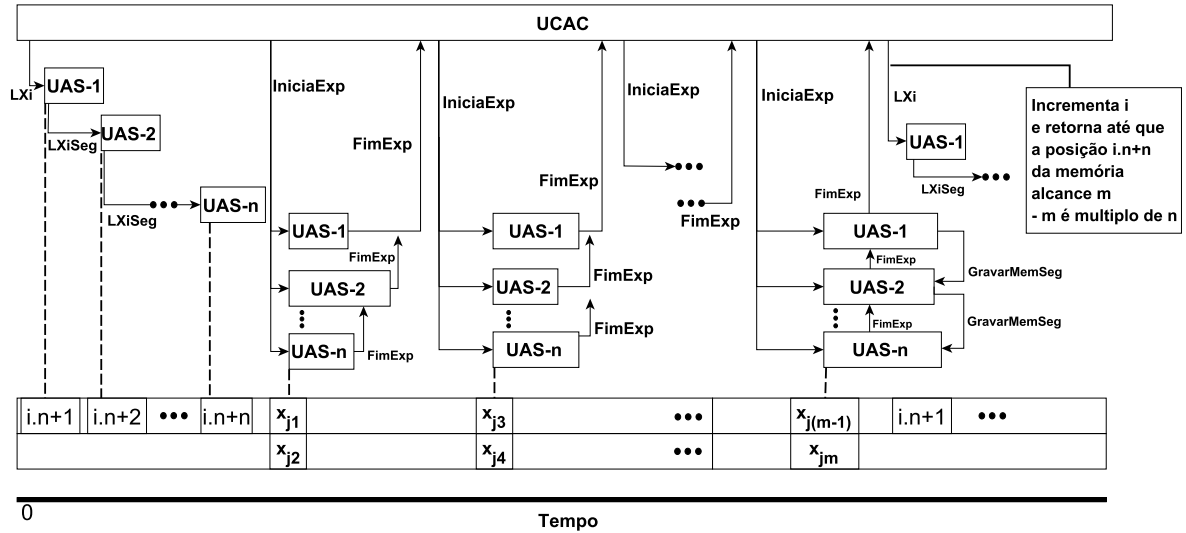


Figura 39: Ilustração da escalabilidade da arquitetura

Após a primeira distribuição de um valor  $x_i$  para cada UAS, o controle em UCAC inicia o cálculo exponencial em todos os componentes UAS ao mesmo tempo, enviando o sinal **IniciaExp**, como ilustrado na Figura 39. Neste momento o primeiro par de dados  $x_j$  já está disponível no barramento BD.

Ao final do cálculo, com o sinal **FimExp**, é iniciada em cada UAS a soma dos dois fatores  $e^{-v}$  calculados. O resultado desta soma será acumulado nos cálculos exponenciais seguintes. Este processo de soma independente do controle em UCAC e não está ilustrado. O sinal **FimExp** é passado de cada UAS para a anterior na arquitetura, sendo que o controle em UCAC só reconhece que o cálculo de um fator  $e^{-v}$  terminou quando este houver terminado em todas as unidades.

A Figura 39 ilustra intervalos de tempo diferentes para o cálculo exponencial em cada UAS, representado pela largura do componente em cada instante. Essa diferença pode ocorrer, principalmente, em função do número de ciclos necessários para enquadrar as frações em cada cálculo. Também tem esse efeito o número de ciclos necessários para encontrar os coeficientes em cada unidade EXP.

Logo após, o cálculo de dois novos fatores  $e^{-v}$  do somatório da Equação 8 é iniciado nos componentes UAS (sinal **IniciaExp**), já com novos dados  $x_j$  em BD. Com isto, os  $n$  componentes UAS implementados calculam em paralelo o potencial de  $n$  pontos do conjunto de amostras.

A ilustração da Figura 39 termina quando todos os valores  $x_j$  já foram disponibilizados e o cálculo do potencial nos  $n$  primeiros pontos da memória foi concluído. Neste momento cada UAS envia um sinal **GravarMemSeg** para a unidade subsequente. Este é enviado como sinal **FimSomaAnt** para o UAS seguinte. Como pode ser visto na Figura 10, o sinal **FimSomaAnt**, junto com o sinal **FimSoma**, habilitam o sinal **GravarMem** para a gravação do dado contido na saída Pi na memória de potenciais MP.

Assim como o sinal **LXiSeg**, o sinal **GravarMemSeg** é habilitado pelo controle de UAS alguns ciclos após receber o sinal **GravarMem**. Isto garante que os dados no barramento BP, para a gravação na memória MP, apareçam na ordem correta. Ou seja, os componentes UAS mais próximos de UCAC gravarão seus dados primeiro, mantendo a ordem de valores potenciais na memória MP correspondente a ordem das amostras  $x_i$  na memória MD. O sinal **GravarMemSeg** garante também que não haja conflito para a gravação da memória quando dois componentes UAS terminarem o somatório no mesmo intervalo de tempo.

Seguindo a ilustração da Figura 39, o valor  $i$  é incrementado em uma unidade e o processo se repete até que todos os pontos tenham o seu potencial calculado.

A primeira UAS faz a verificação do maior valor potencial dentre os calculados por ele e pelas outras UAS, registrando também o índice desse valor máximo, assim como apresentado na Seção 4.3. A primeira UAS também opera no processo de redução de potencial.

#### 4.4.2 Simulações

Esta seção apresenta janelas de simulação da arquitetura escalada com quatro unidades UAS. Os dados são os mesmos da simulação apresentada na Seção 3.5. A máquina de estados da Seção 3.4 dá os detalhes. Na Figura 40, após o sinal para início, o controle em UCAC faz a distribuição dos valores  $x_i$  armazenados na memória MD pelas quatro unidades UAS.

Dois dos quatro sinais de carga dos registradores Xi são mostrados: **LXi** do controle para a primeira UAS e **LXiSeg**, da primeira UAS para a segunda. Após a carga do registrador Xi nas quatro UAS, o sinal **IniciaExp** é enviado. Neste momento o barramento duplo BD passa a fornecer os valores dos endereços 1 e 2. Estes são os dois valores  $x_j$  que serão usados pelos componentes EXP das quatro UAS para calcular o valor exponencial  $e^{-v}$ .



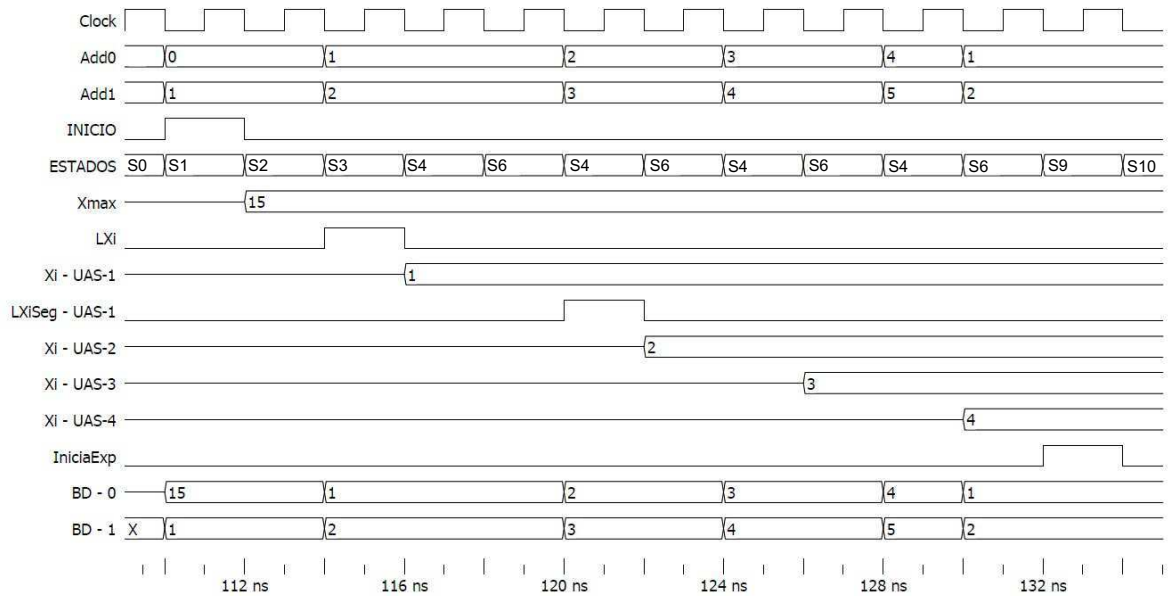


Figura 40: Início da simulação

A Figura 41 apresenta o fim do cálculo exponencial em momentos diferentes nos quatro componentes UAS, mostrando que um novo cálculo só é disparado pelo controle principal (sinal *IniciaExp*) quando todos terminaram.

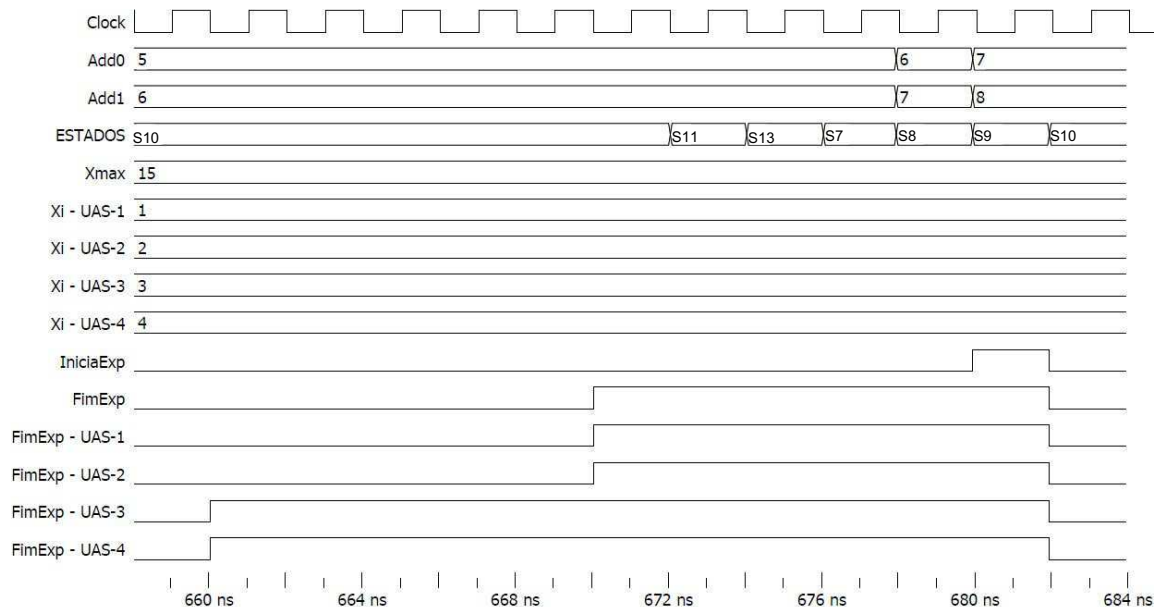


Figura 41: Fim do cálculo exponencial nas UAS

A Figura 42 apresenta o momento em que o somatório termina nos quatro componentes UAS. Pode-se verificar que a ordem de término foi UAS-3, UAS-4, UAS-1 e UAS-2. Mesmo assim

a gravação dos dados manteve a ordem original, seguindo o que foi explicado na Seção 4.4.1: a gravação dos dados de uma UAS só ocorre quando a UAS anterior na arquitetura concluiu a gravação.

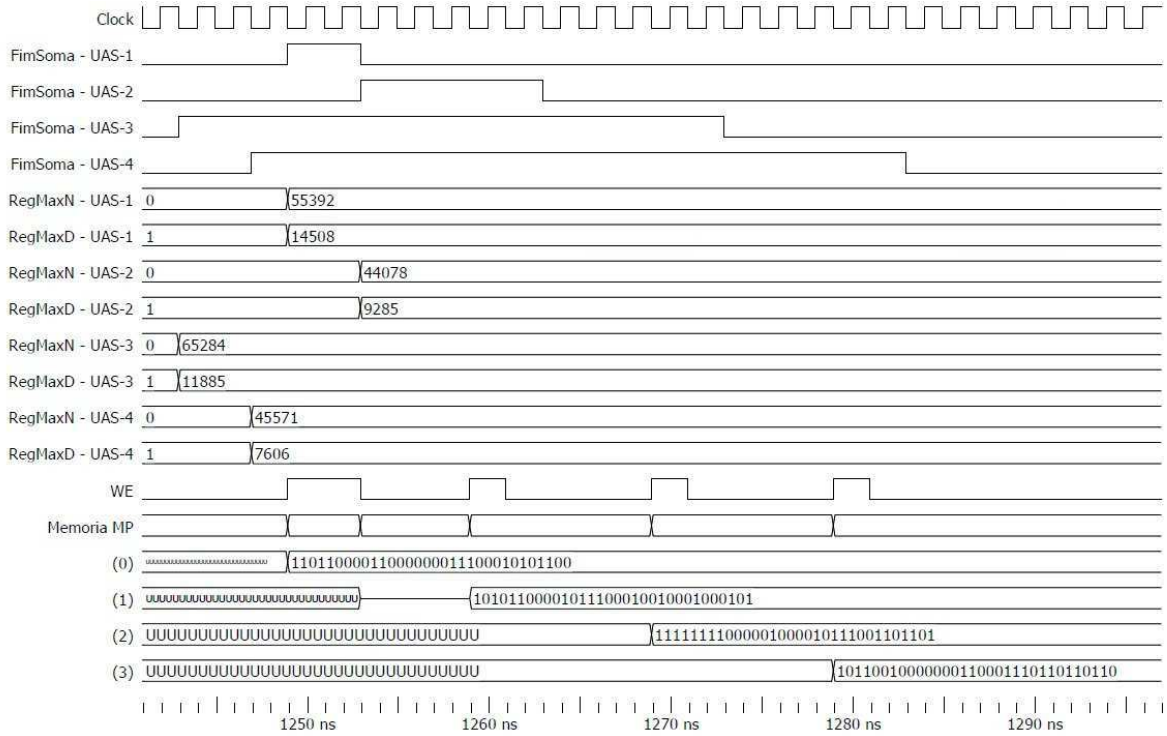


Figura 42: Fim do somatório e gravação de resultados na memória

## 4.5 Considerações Finais do Capítulo

Este capítulo descreveu os detalhes da arquitetura proposta para a implementação do algoritmo para agrupamento subtrativo e a modelagem das operações aritméticas realizadas pela arquitetura. Foi descrita a unidade para agrupamento subtrativo UAS e mostrado os componentes elementares que a compõe, tais como multiplexadores, registradores, deslocadores, somadores e multiplicadores. Grande parte da descrição se apoiou no detalhamento dos controladores dos seus componentes EXP e SOMADOR, que permitem que o algoritmo seja processado pelo *hardware*.

Também foi descrita neste capítulo a escalabilidade da arquitetura. Esta permite que mais componentes UAS atuem em paralelo para o cálculo do valor potencial das amostras, diminuindo proporcionalmente o número de ciclos para a execução do algoritmo de agrupamento subtrativo e fornecendo maior flexibilidade e melhor desempenho à implementação.

---

O próximo capítulo avalia os resultados e o desempenho da arquitetura proposta no processamento do algoritmo de agrupamento subtrativo. Descreve também a implementação em FPGA desta arquitetura, visando a avaliação da área utilizada pela arquitetura nas diferentes configurações e dando a base para o projeto de um sistema portátil que execute o algoritmo em *hardware*.

## Capítulo 5

# IMPLEMENTAÇÃO E RESULTADOS

**E**STE capítulo avalia os resultados e o desempenho da arquitetura proposta no processamento do algoritmo de agrupamento subtrativo.

Os dados usados para testar o algoritmo no MatLab (MATHWORKS, 1999) na Seção 2.3 foram usados em uma simulação no *ModelSim* (MENTOR-GRAPHICS, 2011), para avaliação dos resultados, mostrados na Seção 5.1. Nesta mesma seção os dados de outros radionuclídeos foram testados. A Seção 5.2 avalia o desempenho do processamento do algoritmo no *hardware* proposto em comparação com o processamento em software embarcado em microcontrolador. Na Seção 5.3 será avaliada a relação entre o aumento na área ocupada na FPGA, com um número maior de componentes UAS (unidades de agrupamento subtrativo) na arquitetura, e o aumento na velocidade de processamento. A Seção 5.4 mostra resultados de equipamentos de radiação testados (BLACKADAR et al., 2006). A Seção 5.5 traz as considerações finais do capítulo.

### 5.1 Resultados de Simulação

A Seção 2.3 apresentou o teste com o algoritmo de agrupamento subtrativo no MatLab, utilizando os dados mostrados na Figura 8. Esta figura apresenta o espectro de energia de uma fonte radioativa contendo Cs-137 e Co-60, onde o eixo horizontal corresponde aos canais de um ADC e o eixo vertical ao número de ocorrências acumuladas em cada canal. Estes dados, convertidos em dados unidimensionais, foram processados pela ferramenta *subclust* do MatLab, com o resultado sendo mostrado na mesma Figura 8 através da marca circular. Este conjunto de dados possui 1024 amostras que representam os valores de energia, convertidos em canais de um ADC, que formam este espectro.

Nesta seção será mostrado o resultado da aplicação do mesmo conjunto de dados na simulação da arquitetura proposta, com um componente UAS, no *ModelSim*. Os momentos

de maior importância com relação aos valores computados serão destacados. Os detalhes da simulação da arquitetura foram apresentados nos Capítulos 3 e 4.

A Figura 43 mostra o início da simulação com o carregamento dos registradores cujos os dados serão usados no primeiro cálculo exponencial. São carregados os registradores  $X_{max} = 2800$  e  $X_i = 1324$ . Os dois valores  $x_j$ , 1324 e 2668, um para cada componente EXP, estão disponíveis no barramento duplo BD (sinais BD-0 e BD-1). O cálculo nos componentes EXP é iniciado com o sinal *IniciaExp*.

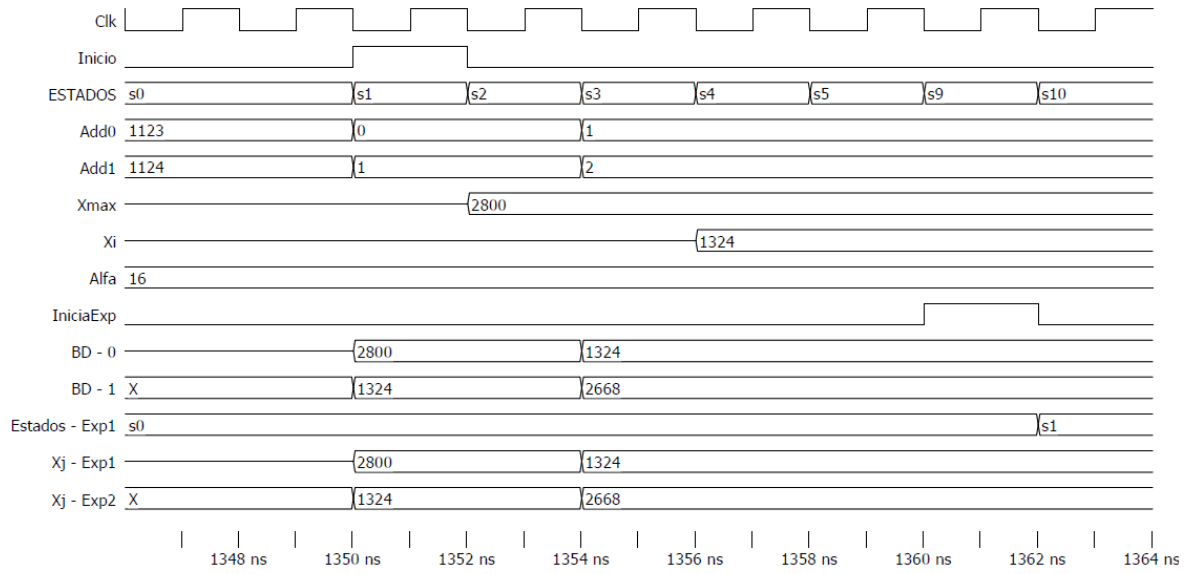


Figura 43: Início do cálculo exponencial na simulação

Os dois componentes EXP realizam o cálculo exponencial  $e^{-v}$ , onde  $v$  é dado pela Equação 10. Deste modo, o valor fornecido pelo dois componentes EXP deverá ser dado respectivamente pelas Equações 17 e 18.

$$e^{\frac{-\alpha||x_j-x_i||^2}{(x_{max})^2}} = e^{\frac{-16||1324-1324||^2}{(2800)^2}} = 1 \quad (17)$$

$$e^{\frac{-\alpha||x_j-x_i||^2}{(x_{max})^2}} = e^{\frac{-16||2668-1324||^2}{(2800)^2}} = e^{-3,6864} = 0,0251 \quad (18)$$

A Figura 44 apresenta o final do cálculo exponencial na simulação, com o resultado do componente EXP1 em Np1 e Dp1 (numerador e denominador) e de EXP2 em Np2 e Dp2, que representam as frações que serão usadas no componente SOMADOR. Verifica-se que os valores dessas frações não são exatamente iguais aos encontrados nas Equações 17 e 18. Isso ocorre em virtude da aproximação por polinômios quadráticos e da representação dos valores em forma de fração, utilizadas pela arquitetura, como explicado nos Capítulos 3 e 4.

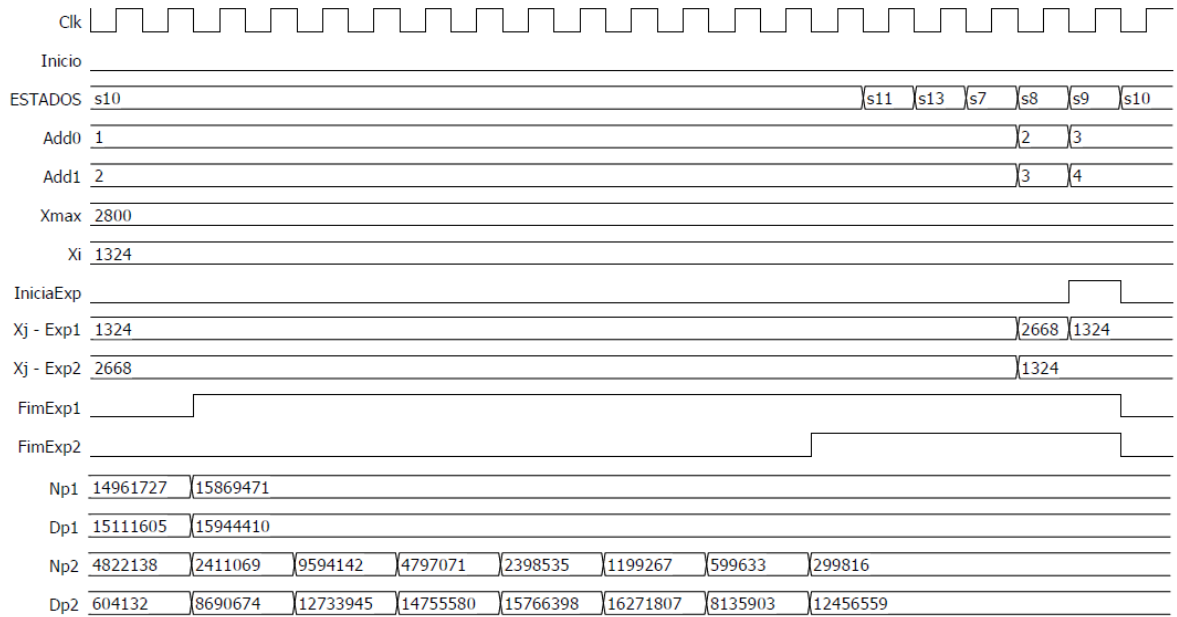


Figura 44: Final do cálculo exponencial

A Figura 45 apresenta o final da primeira soma no somatório para o cálculo do valor potencial, dado pela Equação 8. Os registradores RegAuxN/RegAuxD guardam o resultado da soma das frações  $Np1/Dp1$  e  $Np2/Dp2$ , carregados no estado S1 do controle do componente SOMADOR.

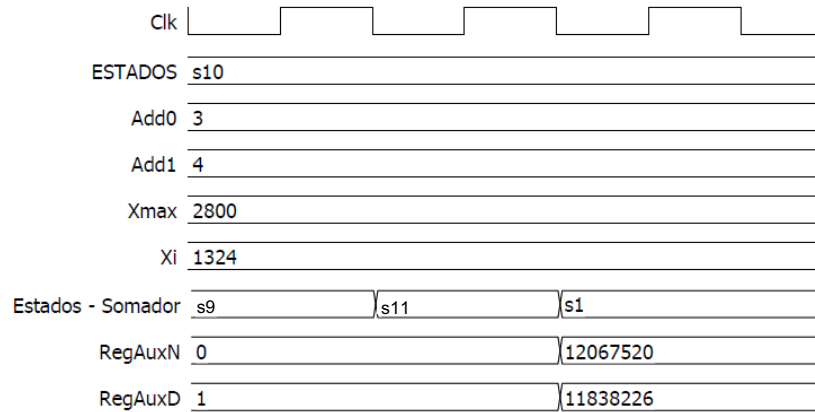


Figura 45: Final da primeira soma

A Figura 46 mostra o resultado do primeiro valor potencial calculado, o potencial do ponto  $x_i = 1324$ , o primeiro endereço da memória MD. Este resultado é obtido depois que os 1024 fatores exponenciais foram calculados, somados e acumulados como visto nos passos anteriores. O resultado em forma de fração é passado de RegAuxN/RegAuxD para

RegMaxN/RegMaxD e o índice da memória MD (RegIndice) é incrementado. O valor em RegAuxN/RegAuxD também é gravado na memória MP.

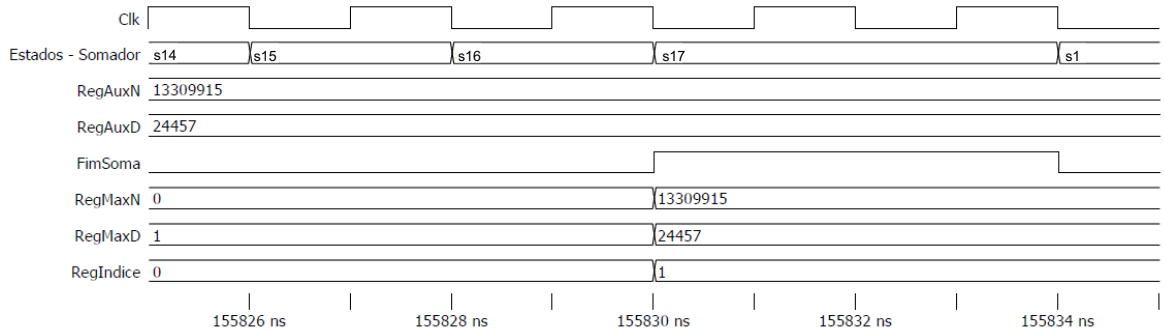


Figura 46: Final do primeiro somatório e do cálculo do primeiro potencial

A partir deste momento um novo valor potencial começa a ser acumulado em RegAuxN e RegAuxD. Este é o valor potencial do ponto  $x_i = 2668$ , o segundo endereço da memória MD. A Figura 47 mostra o final do somatório para o cálculo desse valor potencial. Como este valor potencial em RegAuxN/RegAuxD é menor que o valor anterior, registrado em RegMaxN/RegMaxD, o valor nestes dois últimos registradores é mantido depois da comparação.

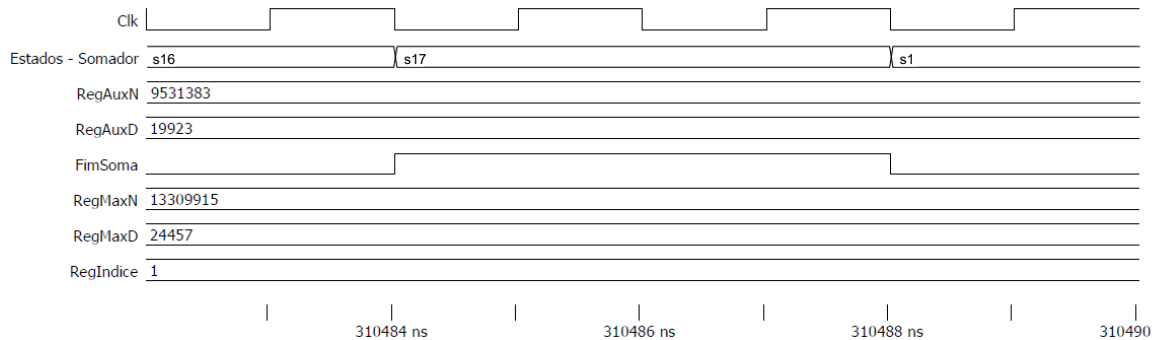


Figura 47: Final do segundo somatório

Esse processo se repete até que o valor em  $x_i$  seja o último endereço na memória MD. Neste momento todos os valores potenciais foram calculados e o registrador RegIndice guarda o índice da memória MD do potencial máximo. Esse índice, o valor 217 no registrador RegIndice da Figura 48, é o índice do primeiro centro de agrupamento obtido e corresponde ao valor 1324 que é carregado da memória MD no registrador Xindex para ser usado no processo de redução.

Verifica-se que o ponto de maior potencial encontrado corresponde, como esperado, ao valor do primeiro pico do espectro, característico do radionuclídeo Cs-137, como visto na

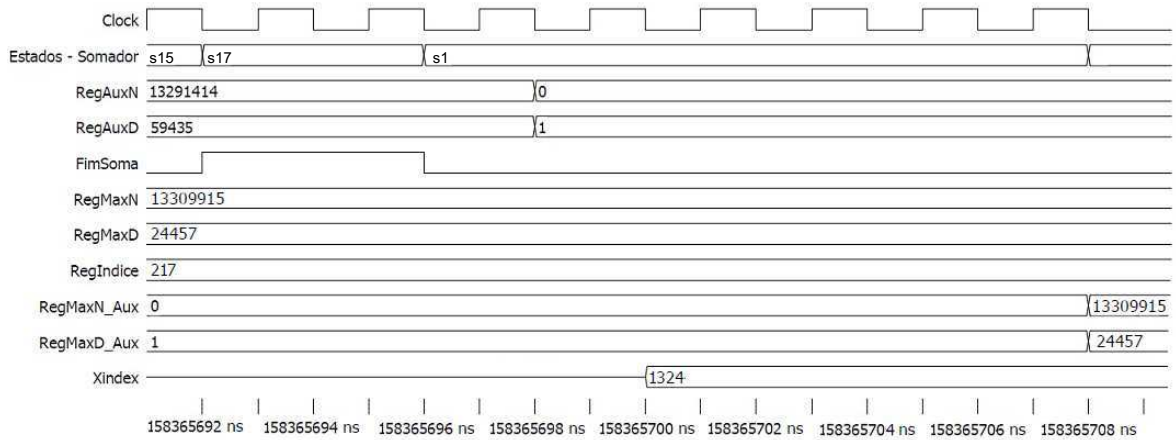


Figura 48: Final do cálculo do último potencial

Tabela 1: Radionuclídeos testados			
Radionuclídeo	Energias características (keV)	Canais correspondentes	Agrupamentos encontrados (canais)
Bário-133 (Ba-133)	81 e 356	162 e 712	161, 711 e 160
Európio-152 (Eu-152)	122, 245 e 344	244, 490 e 668	243 e 490
Bário-133 (Ba-133) e Césio-137 (Cs-137)	81, 356 e 662	162, 712 e 1324	165, 716 e 1326
Cobalto-60 (Co-60)	1173 e 1333	815 e 919	469, 138, 837 e 857

Seção 2.3. Verifica-se também que o tempo para processar esse conjunto de dados foi de aproximadamente 160 ms, considerando um ciclo de 2 ns na simulação, o que resulta em um processamento em aproximadamente 80 milhões de ciclos.

Conjuntos de dados referentes a outros radionuclídeos foram testados utilizando a simulação no *ModelSim* e apresentando resultados coerentes. A Tabela 1 apresenta os testes realizados. A Figura 49 apresenta os espectros correspondentes.

Os agrupamentos encontrados da Tabela 1 são apresentados na ordem em que foram descobertos pelo algoritmo. No primeiro item, referente ao teste com o radionuclídeo Bário-133, verifica-se que os dois picos principais são bem aproximados nos canais encontrados. Para o primeiro pico, no canal 162, são encontrados os canais próximos 161 e 163, respectivamente o primeiro e terceiro centro de agrupamento encontrado.

No segundo item, o teste com o radionuclídeo Európio-152, dois dos três picos principais de energia característica são encontrados, sendo isto suficiente para caracterizar o radionuclídeo. No teste com o terceiro item verifica-se que os três canais encontrados se aproximam dos três picos que serviriam para caracterizar os dois radionuclídeos presentes na amostra. Isto



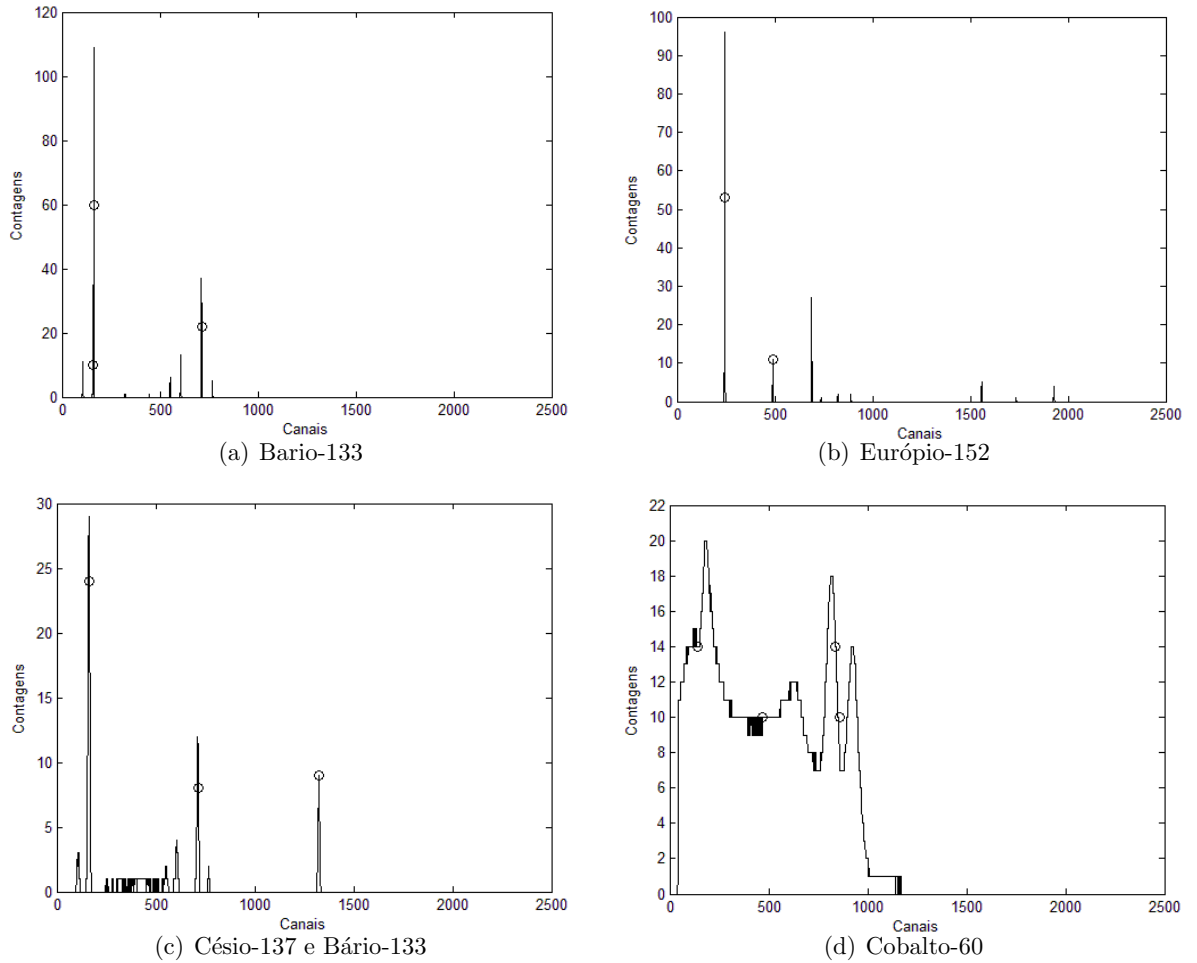


Figura 49: Espectros testados

demonstra a capacidade do algoritmo, processado pela arquitetura proposta, de encontrar mais de um radionuclídeo presente em uma amostra.

O teste com Co-60, o quarto item, foi realizado com dados de um sistema real de detecção baseado em detector de Iodeto de Sódio. Os dois valores de energia característica desse radionuclídeo são os dois últimos picos da Figura 49–(d). Nota-se que neste espectro estes picos não são os valores de maior contagem. Isto pode ocorrer com alguns sistemas de detecção. As causas podem ser a pobre resolução de energia, inerente a alguns detectores, a forma de acondicionamento do radionuclídeo, ruídos do circuito de amplificação, dentro outros, o que mostra a dificuldade de identificar radionuclídeos em alguns casos.

Neste caso o algoritmo não conseguiria distinguir o canal de energia do Co-60 como o primeiro centro de agrupamento, como realmente aconteceu. Verifica-se que o primeiro centro de agrupamento descoberto, o canal 469, é um valor intermediário entre os picos, também com concentração de contagens. Isto é um resultado coerente com o algoritmo, considerando que

o raio do agrupamento usado abranja uma região com contagens significativas em todos os canais, como ocorre para esse espectro. No entanto esse é um valor que não deveria ser usado para classificar um radionuclídeo, pois não representa um valor de energia característica da amostra. Caso esse valor se aproxime de um valor de energia característica de outro radionuclídeo, não presente na amostra, ocorreria provavelmente uma classificação *falso positivo* de um radionuclídeo. O mesmo ocorre para o segundo canal encontrado, o valor 138. Somente o terceiro valor encontrado, o canal 837, se aproxima de um canal de energia característica para o Co-60, o valor do pico em 815. Mesmo essa aproximação sendo bem menor do que nos casos anteriores, o resultado serviria para identificá-lo com algum grau de certeza.

Variações nos parâmetros  $\alpha$  e  $\beta$  das Equações 8 e 9, dadas no Capítulo 3, podem ser usadas para ajustar o algoritmo aos dados recebidos e a um determinado conjunto de detecção. Porém, é importante que o sistema de detecção seja o mais favorável possível ao funcionamento correto do algoritmo processado pela arquitetura. Isto significa principalmente possuir boa resolução de energia e baixo ruído.

O caso do teste com Co-60, mostrado no quarto item da Tabela 1, demonstra que a arquitetura proposta é robusta para descobrir os canais de energia característica, mesmo com dados pouco favoráveis ao algoritmo, porém ao risco de levar à identificação de *falsos positivos*. A maior possibilidade de identificação de *falsos positivos*, em relação a identificação de *falsos negativos*, é uma característica da arquitetura que traz vantagens. A ocorrência de *falsos positivos* é conservadora, permitindo que o erro seja corrigido em análise posterior. No entanto, a ocorrência de *falsos negativos* dificulta a correção do erro, levando a não identificação do radionuclídeo. Neste caso, a passagem de elementos radioativos em um sistema de vigilância, por exemplo, poderia ocorrer sem a possibilidade de correção posterior. A Seção 5.4 mostrará que em sistemas de identificação de radioisótopos já desenvolvidos, a ocorrência de *falsos positivos* e *falsos negativos* é praticamente a mesma.

## 5.2 Resultados de Processamento

Os algoritmos de agrupamento são, em geral, intensos consumidores de tempo de processamento em virtude das iterações que se multiplicam com o aumento no número de amostras. Nesta seção será mostrada a aplicação do algoritmo de agrupamento subtrativo em um processador MicroBlaze<sup>TM</sup> (XILINX, 2005a) para avaliar o tempo de processamento em software embarcado. Essa avaliação serviu para verificar o ganho de desempenho que uma solução em *hardware*, como a proposta nesta dissertação, traz à execução de algoritmos de agrupamento.

Tabela 2: Comparação no número de ciclos para processamento

		Espectros			
		Ba-133	Eu-152	Cs-137 e Ba-133	Co-60
Número de amostras		412	520	775	9654
		Número de ciclos $\times 10^3$			
Microblaze		2739587,467	4364128,325	9690870,878	-
Hardware	1 UAS	12959,789	20644,026	45855,467	7115449,045
	2 UAS	6479,890	10322,017	22927,735	3557724,524
	3 UAS	4319,931	6881,344	15285,156	2371816,349
	4 UAS	3239,949	5161,007	11463,867	1778862,262

O MicroBlaze™ (XILINX, 2005a) é um processador, otimizado para implementação em dispositivos FPGA, da empresa *Xilinx*. Este é um *soft-processor*, ou seja, é um processador construído em linguagem de descrição de *hardware*. A sua arquitetura é de 32 bits, com um conjunto de instruções reduzidas (RISC - *Reduced Instruction Set Computing*), podendo ser sintetizado e implementado em FPGA.

Para implementar o processador MicroBlaze™ em uma FPGA é necessário utilizar a ferramenta XPS da *Xilinx* (*Xilinx Platform Studio*) (XILINX, 2008). Com esta ferramenta é possível configurar o processador com os periféricos necessários e desenvolver um software que será embarcado no processador. Este software pode ser desenvolvido em linguagem C.

Nesta implementação foram utilizados os quatro conjunto de amostras testados na Seção 5.1, que representam os espectros de radionuclídeos da Figura 49. O algoritmo de agrupamento subtrativo foi realizado em C, com as iterações necessárias para calcular o potencial de cada uma das amostras fornecidas. A implementação foi realizada em FPGA *Spartan 3E* (XILINX, 2009) do kit de desenvolvimento *Starter kit* (XILINX, 2011), com um circuito contador para a verificação do número de ciclos de *clock* usados na execução do algoritmo.

Os mesmos conjuntos de amostras foram usados em simulações utilizando o *ModelSim* em quatro situações, variando o número de unidades UAS de um a quatro. Os resultados com o número de ciclos para processamento dos agrupamentos são mostrados na Tabela 2. Esta tabela apresenta o número de amostras/pontos fornecidos por cada espectro. O teste com o conjunto maior de amostras (Co-60) não foi processado pelo MicroBlaze™.

Os resultados mostram que a tarefa de processar os dados de agrupamento em *software* embarcado pode comprometer uma aplicação que exija respostas mais rápidas como, por exemplo, o caso de instrumentos portáteis. Pode-se constatar também que o uso de *hardware* digital dedicado oferece um ganho em desempenho que pode viabilizar aplicações com grande demanda de processamento. Outra informação que pode ser retirada dos resultados se refere a

vantagem de se construir uma arquitetura de *hardware* com flexibilidade para a escalabilidade, permitindo que seja diminuído o tempo de processamento de forma inversamente proporcional ao acréscimo de unidades de cálculo em paralelo. A Seção 5.3 avalia essa melhora no desempenho em função do aumento na área ocupada.

## 5.3 Implementação em FPGA

Esta seção verifica a área ocupada e a frequência máxima de operação de uma FPGA *Spartan 3E* para uma avaliação da relação destes fatores com o aumento de desempenho oferecido pela escalabilidade.

A FPGA *Spartan 3E* (XILINX, 2009), modelo XC3S500E-4FG320, possui 232 pinos de entrada e saída (IOB) e 4656 blocos lógicos elementares, chamados de *slice*. Na FPGA *Spartan 3E* cada conjunto de quatro *slices* forma um CLB, os blocos lógicos configuráveis. Os *slices* são formados por duas *look-up tables* (LUTs) que podem implementar qualquer lógica combinacional de quatro variáveis de entrada e uma saída. Além da LUTs, cada *slice* contém também dois *flip-flops* que atuam como circuitos armazenadores de bits.

O kit de desenvolvimento utilizado, o *Starter kit* (XILINX, 2011), possui, além da FPGA, um conjunto de interfaces (USB, RS-232 e VGA) e características adicionais como mostrador de cristal liquido, chaves e leds.

A implementação da arquitetura foi realizada com a ferramenta *Xilinx ISE* (XILINX, 2012a). Para a avaliação da área ocupada foi considerado o número de *slices* ocupados e o número de LUTs e *flip-flops* utilizados. A Figura 50 apresenta um gráfico de barras com estes números e com valores percentuais de ocupação em função do aumento no número de unidades UAS na arquitetura.

A ferramenta *Xilinx ISE* também informa a frequência máxima de operação da arquitetura sintetizada e mapeada no dispositivo. Neste caso foi informada uma frequência máxima de operação de 68,213 MHz para um componente UAS e 39,742 MHz para dois componentes.

Para a FPGA *Spartan 3E* não foi possível a implementação de mais de duas unidades em paralelo em função do número de LUTs ter ultrapassado o número disponível no dispositivo. Verifica-se que a implementação de componentes UAS em paralelo provoca um aumento na área ocupada em proporção maior do que o aumento no desempenho. Este fato mostra que a escalabilidade, embora melhore o desempenho e permita maior flexibilidade para a implementação, impõe um comprometimento de área ocupada no dispositivo. Contudo, implementações

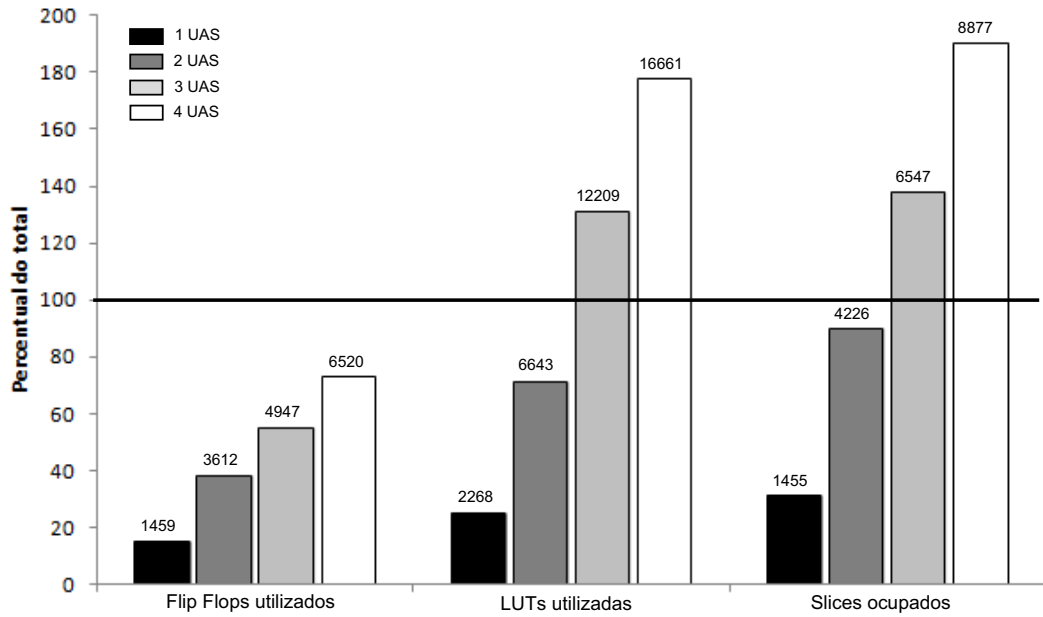


Figura 50: Área ocupada em função do número de UAS na *Spartan 3E*

em famílias de FPGA com maior disponibilidade de blocos lógicos podem obter ganho desta característica da arquitetura proposta.

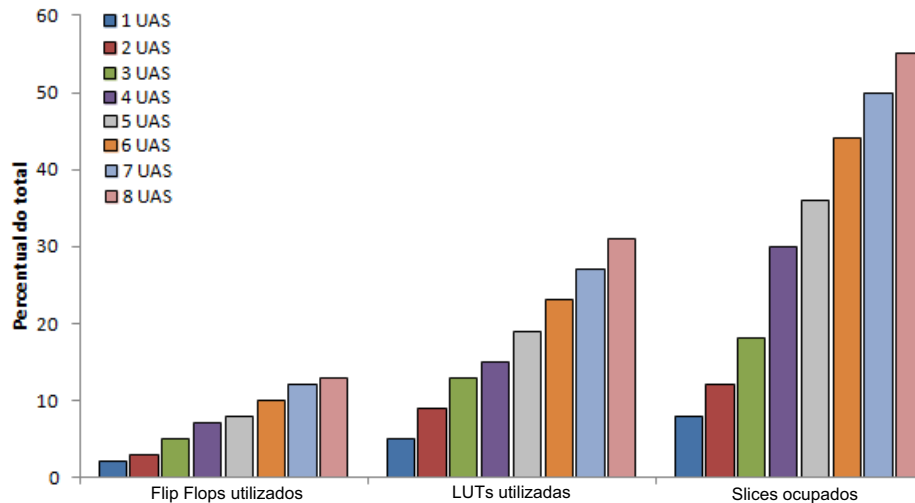
Para a aplicação proposta neste trabalho, a utilização de memória com 1024 posições é suficiente para que os dados de energia dos radionuclídeos sejam caracterizados. Isto considerando o uso de distintos detectores de radiação, que podem necessitar de números de amostras de energia que variam em função das suas características de sensibilidade e resolução. Na implementação realizada, a memória RAM foi implementada usando blocos de memória selecionáveis, os blocos *SelectRam*<sup>TM</sup> (XILINX, 2005b). Com estes blocos, até 72 kbits (9 kBytes) de memória podem ser configurados e instanciados em uma arquitetura, sem comprometer o número de *slices* ocupados e o número de LUTs e *flip-flops* utilizados. Portanto, a implementação pode usar distintos tamanhos de memórias, até o limite de 9 kBytes, sem que ocorra alteração no número de blocos lógicos utilizados. Com 1024 posições de memória de 16 bits cada, a memória MD utiliza 2 kBytes. A memória MP, onde cada palavra terá 32 bits neste caso, ocupará mais 4 kBytes, em um total de 6 kBytes.

Para fazer a verificação de área ocupada em famílias de FPGA mais modernas, a arquitetura foi implementada, com a mesma ferramenta *Xilinx ISE* utilizada na implementação da FPGA *Spartan 3E*, usando desta vez a FPGA *Virtex 6*. A *Virtex 6* (XILINX, 2012b), modelo xc6vlx75t-3ff484, escolhida para essa implementação, pertence a mais nova família de FPGAs para arquiteturas digitais de alto desempenho, com 744986 *slices* contendo 4 LUTs e 8 *flip-*

Tabela 3: Área ocupada e frequência máxima de operação em FPGA *Virtex 6*

	<i>Flip Flops</i> utilizados	LUTs utilizadas	<i>Slices</i> ocupados	Frequência máxima (MHz)
1 UAS	1809	2400	993	211,8
2 UAS	3264	4386	1424	186,9
3 UAS	4724	6129	2117	177,4
4 UAS	6181	7446	3581	157,4
5 UAS	7643	9179	4258	154,2
6 UAS	9141	10905	5190	149,4
7 UAS	10599	12674	5885	149,2
8 UAS	12054	14479	6505	149,0

*flops*, com cada LUT possuindo 6 entradas e uma saída. A Figura 51 mostra um gráfico com a ocupação de área nesta FPGA em função do número de componentes UAS em paralelo.

Figura 51: Área ocupada em função do número de UAS *Virtex 6*

Na Tabela 3 é apresentado o número de *slices* ocupados e o número de LUTs e *flip-flops* utilizados, apresentados na Figura 51, além da frequência máxima de operação que pode ser usada nesta implementação.

A implementação no kit de desenvolvimento utilizado, com a FPGA *Spartan 3E*, permitiu observar que o *hardware* desenvolvido possui desempenho compatível com o informado pelas simulações realizadas. O número de ciclos de *clock* necessários para calcular os valores potenciais durante o processamento do algoritmo foi o mesmo observado nas simulações no *ModelSim*.

## 5.4 Comparação com outras soluções

Existem no mercado cerca de uma dezena de equipamentos portáteis para obter o espectro e fazer a identificação automática de radionuclídeos. O número de radionuclídeos que podem ser identificados por estes varia, conforme o modelo, entre dez e setenta diferentes tipos. Como foi visto no Capítulo 1, os métodos em geral se baseiam na busca no espectro pelos picos de energia característica que correspondem a cada radionuclídeo (KNOLL, 1989). Alguns fatores podem degradar a resposta desses equipamentos, principalmente para uso no campo. Esses fatores são listados abaixo (BLACKADAR et al., 2006):

1. Tempo de aquisição: Alguns equipamentos necessitam de um tempo maior de aquisição para obter uma resposta confiável. Este fator pode ser em decorrência do tipo de detector usado, necessitando de mais tempo os de menor sensibilidade. Também pode ser em decorrência do método usado neste equipamento para fazer a identificação, considerando que algoritmos de busca pelos canais de maior ocorrência de energia podem ser lentos em microprocessadores de propósito geral. Portanto, sem um intervalo de tempo adequado para a aquisição, fator comum em atividades no campo, o desempenho na identificação pode ser prejudicado;
2. Variação na calibração: A relação entre os canais de um ADC e a energia da radiação detectada é feita através de ajustes na calibração. Se essa relação não é ajustada de maneira adequada, conforme o que está registrado na configuração do equipamento, a identificação pode ser não realizada corretamente.
3. Mudanças de temperatura: A maioria dos detectores de radiação são suscetíveis à mudanças de temperatura, comuns em atividades no campo. Isto altera de imediato a calibração do equipamento, exigindo que esta seja refeita com mais frequência.
4. Mudança nos níveis de radiação de fundo: A radiação de fundo é a radiação existente no ambiente, podendo variar em função deste. Apesar de menos frequente, a mudança na radiação de fundo deve ser medida pelo equipamento para que se diminua o fator de ruído do sistema de detecção e a identificação de radionuclídeos seja facilitada.

Esses fatores, por serem de difícil controle, complicam a tarefa de identificar radionuclídeos com equipamentos portáteis, ocasionando casos de *falsos positivos* e *falsos negativos*. Testes em equipamentos (BLACKADAR et al., 2006) realizados com 7 modelos de mercado mostraram que apenas 27 % das medidas realizadas, com 63 fontes radioativas, conseguiram iden-

tificar corretamente o radionuclídeo com maior concentração na amostra. Os casos de *falsos positivos* e *falsos negativos* também ficaram em torno de 27 %. Os equipamentos testados foram: SAM 935 (BERKELEY, 2005), Quantradr Sensor Ranger (QUANTRAD, 1998), Exploranium GR-135 (SAIC, 2009), Exploranium GR-135 CZT (SAIC, 2009), SAIC Radsmart (SAIC, 2000), Bicorn FieldSPEC (BICRON, 2003) e CZT-PP.

A Figura 52 (BLACKADAR et al., 2006) mostra um gráfico com estes resultados, onde *condicionalmente correto* significa que o radionuclídeo com maior concentração na amostra foi identificado, mas após a identificação de um radionuclídeo que não estava presente na amostra. A categoria MD, ou *minor daughter*, significa que um radionuclídeo *filho*, ou seja, um produto do decaimento do radionuclídeo mais presente na amostra foi identificado (ANSI, 2007).

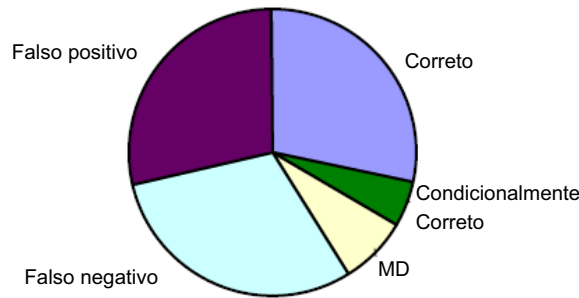


Figura 52: Gráfico de testes realizados com 7 equipamentos

Uma comparação dos dados destes testes com os resultados obtidos pela arquitetura proposta é prejudicada. Isso ocorre por não ser possível saber qual foi o espectro recebido pelo algoritmo destes equipamentos para realizar a identificação. Também a variedade de espectros de radionuclídeos testados foi menor no caso desta dissertação. No entanto, nos testes mostrados neste capítulo, somente em uma ocasião ocorreu um resultado *condicionalmente correto*, na identificação do CO-60 mostrado na Figura 49-(d). Neste caso o primeiro agrupamento identificado, o valor 469, poderia ter sido identificado como o radionuclídeo Sódio-22 (Na-22). Nos quatro demais casos testados a identificação foi correta, inclusive, seguindo a obrigatoriedade dada pelo padrão americano (ANSI, 2007), conseguindo realizar a identificação de dois radionuclídeos presentes na mesma amostra.

A Tabela 4 mostra estes resultados comparados com os sete equipamentos citados (BLACKADAR et al., 2006), identificados de 1 a 7, para os mesmos radionuclídeos testados. Os testes com outros equipamentos não foram realizados com dois radionuclídeos na mesma amostra, Ba-133 e Cs-137 ou Cs-137 e Co-60, como foi realizado com a arquitetura proposta. Isto está indicado por 'NT'. O '✓' informa que houve a identificação correta pelo equipa-



mento. ‘CC’ identifica o caso *condicionalmente correto* e X um erro de *falso positivo* ou *falso negativo*.

Tabela 4: Comparação de resultados com equipamentos de identificação de radionuclídeos

Espectros	1	2	3	4	5	6	7	Arquitetura Proposta
Ba-133 e Cs-137	NT	NT	NT	NT	NT	NT	NT	✓
Cs-137 e Co-60	NT	NT	NT	NT	NT	NT	NT	✓
Ba-133	x	✓	✓	✓	✓	✓	✓	✓
Eu-152	✓	✓	x	✓	✓	✓	✓	✓
Co-60	✓	✓	x	✓	✓	✓	✓	CC
Na-22	✓	✓	x	x	✓	x	✓	✓

## 5.5 Considerações Finais do Capítulo

Neste capítulo foram apresentados alguns resultados da simulação da arquitetura proposta. Foi visto que esta realiza adequadamente os cálculos para a obtenção dos agrupamentos, conforme o algoritmo determina. A avaliação do desempenho da arquitetura, em comparação com uma solução em *software* embarcado e na implementação em FPGA, mostrou que o desenvolvimento de uma solução em *hardware* traz benefícios, permitindo que aplicações com uso intenso de recursos computacionais sejam implementadas em soluções portáteis.

O próximo capítulo conclui o trabalho e propõe outras aplicações que podem utilizar, com melhorias à arquitetura computacional desenvolvida nesta dissertação, a ideia de agrupamento de dados em *hardware*.

## Capítulo 6

# CONCLUSÃO E TRABALHOS FUTUROS

NESTA dissertação foi apresentada uma arquitetura de *hardware* que atua no processamento do algoritmo de agrupamento subtrativo. O objetivo principal foi desenvolver em *hardware* reconfigurável uma arquitetura que pudesse ser empregada na identificação de radionuclídeos, através dos dados de energia  $\gamma$  provenientes de uma fonte radioativa.

Esta arquitetura foi modelada em VHDL buscando o maior paralelismo possível para a computação aritmética e a possibilidade de escalabilidade.

### 6.1 Resumo e Conclusão

A arquitetura proposta nesta dissertação foi concebida para executar em *hardware* um algoritmo de agrupamento que fornecesse um resultado correspondente as energias características de radionuclídeos presentes em amostras sob estudo. O algoritmo escolhido, de agrupamento subtrativo, foi testado com o software MatLab e apresentou resultados coerentes na identificação dos valores de energia  $\gamma$  característica. Portanto, o objetivo desde o início foi implementar este algoritmo em *hardware* de forma que a execução fosse otimizada e mais rápida possível.

Já se sabia da observação da execução do algoritmo no MatLab, e do estudo do próprio algoritmo, que a execução deste exige muitos cálculos iterativos, o que demanda tempo para a conclusão da execução. Portanto, o início do projeto foi planejado para testar soluções que visavam a simplificação do *hardware* em desenvolvimento.

A primeira a ser testada, o uso da representação numérica de números reais em forma de frações de inteiros, já havia sido testada em *hardware* reconfigurável com sucesso (SILVA; NEDJAH; MOURELLE, 2009). A segunda simplificação para implementação do *hardware*, a aproximação da exponencial  $e^{-v}$  por um polinômio de segundo grau, já havia sido testada em

outro trabalho (SILVA, 2010), mas necessitou ser aprimorada com a introdução de uma faixa adicional de aproximação para  $v \in [0, 1[$ .

A possibilidade de executar o algoritmo com operações aritméticas de soma, subtração e multiplicação foi crucial para definir as estratégias para tornar o *hardware* o mais paralelo possível no cálculo dessas operações. Os distintos controladores, baseados em máquinas de estado, buscaram a implementação desse paralelismo à arquitetura. Isto permitiu diminuir os ciclos de *clock* para execução do algoritmo e obter os resultados de forma otimizada.

Outra decisão que trouxe flexibilidade e ganho de desempenho à arquitetura foi o desenvolvimento da escalabilidade, que se resume na possibilidade de implementar outras unidades de cálculo, as UAS, em paralelo. As simulações e a implementação em FPGA permitiram verificar que esta solução traz um ganho de desempenho que deve ser considerado na implementação do algoritmo, mesmo com o aumento na área ocupada na FPGA. Esta solução também pode ser aplicada no desenvolvimento de outras soluções em *hardware* que necessitem implementar escalabilidade.

Testes em equipamentos (BLACKADAR et al., 2006) já desenvolvidos mostram que os casos de *falsos positivos* e *falsos negativos* possuem taxa de ocorrência semelhante. O uso do método de agrupamento subtrativo trouxe à arquitetura desenvolvida uma característica de maior possibilidade de ocorrência de *falsos positivos* em relação a ocorrência de *falsos negativos*. Essa é uma característica que traz vantagens pois a ocorrência de *falso positivo* é conservadora, permitindo que o erro seja corrigido em análise posterior, ao contrário da ocorrência de *falso negativo*.

Uma melhoria no desempenho pode ainda ser obtida com alterações na arquitetura que permitam que esta não necessite calcular o valor potencial de uma amostra que já foi calculada. Neste caso o *hardware* teria de reconhecer um valor de amostra já calculado e buscar este valor armazenado. Estas alterações poderão ser desenvolvidas em trabalhos seguintes.

Os resultados mostraram que a arquitetura proposta cumpre o objetivo inicial de conceber um *hardware* que desempenhe com eficiência a tarefa de executar o algoritmo de agrupamento subtrativo para a identificação de radionuclídeos. A continuidade deste trabalho será comentada na Seção 6.2

## 6.2 Trabalhos Futuros

A arquitetura de *hardware* proposta nesta dissertação é a parte principal e a base para o desenvolvimento de um equipamento portátil de identificação de radionuclídeos. Para a conclusão

do desenvolvimento deste equipamento, ainda será necessário o desenvolvimento do sistema de detecção da radiação  $\gamma$ , o que inclui no mínimo o detector de radiação, circuitos amplificadores e o conversor analógico digital (ADC – *Analog to Digital Converter*). O ADC terá que ser integrado na arquitetura proposta, preenchendo a memória MD, conforme apresentada no Capítulo 3, com os dados que representam a energia detectada.

Outro sistema, que pode ser baseado em microprocessador de propósito geral, deve ser desenvolvido e integrado ao *hardware* proposto para capturar o resultado deste e verificar, em dados gravados em biblioteca, quais energias identificadas pertencem a que radionuclídeo. Este sistema também deve controlar a apresentação dos dados em mostrador. A conclusão destas tarefas permitirá o depósito de patente do equipamento portátil de identificação de radionuclídeos.

O *hardware* desenvolvido também abre a possibilidade de implementar o algoritmo de agrupamento subtrativo como auxiliar na solução de problemas onde um conjunto de dados precisam ser processados em tempo real. Devido ao seu desempenho muito superior aos das soluções em software, como verificado pelos testes realizados, a aplicação em *hardware* pode auxiliar na tomada de decisão em sistemas dinâmicos, como por exemplo, na validação de sinal em plantas industriais ou nucleares.

A validação de sinal é definida como a identificação, em tempo real, de falhas no processo de medida e a subsequente produção da melhor estimativa para o valor da variável que está sendo monitorada (OLIVEIRA, 2005). Alguns trabalhos (OLIVEIRA, 2005; NING; CHOU, 1992; HINES; WREST; UHRIG, 1997) usam técnicas de agrupamento de dados como parte da realização do processo de validação de sinal. Com adaptações e melhoramentos, o *hardware* desenvolvido poderia ser usado, com vantagens de desempenho, em processos de validação de sinal em tempo real.

# REFERÊNCIAS

ANDREUCCI, R. *Proteção Radiológica - Aspectos Industriais*. [S.l.]: Abendi, 2010.

ANSI. Performance criteria for hand-held instruments for the detection and identification of radionuclides. *ANSI Standard N42.34*, 2007.

BELLINTANI, S. A.; GILI, F. N. *Noções Básicas de Proteção Radiológica*. São Paulo: IPEN/CNEN, 2002.

BERKELEY. *Sam-935 - Surveillance and Measurement System*. [S.l.], 2005. Manual.

BICRON. *FieldSPEC*. [S.l.], 2003. Manual.

BLACKADAR, J. et al. *Evaluation of Handheld Isotope Identifiers*. [S.l.], 2006. Los Alamos.

CHEN, S.; ALAHAKOON, D.; INDRAWAN, M. Building an adaptive hierarchy of clusters for text data. *Computational Intelligence for Modelling, Control and Automation*, p. 7–12, 2005.

CHIU, S. L. A cluster estimation method with extension to fuzzy model identification. *Proc. IEEE Internat. Conf. on Fuzzy Systems*, p. 1240–1245, 1994.

CHUNG, K. C. *Introdução à Física Nuclear*. RJ: EdUERJ, 2001.

CNEN. *Diretrizes Básicas de Proteção Radiológica*. [S.l.], 2011.

EISEN, M. B. et al. Cluster analysis and display of genomewide expression patterns. *PNAS* 95:14863-14868, 1998.

EVERITT, B.; LANDAU, S.; LEESE, M. *Cluster analysis*. London: Arnold, 2001.

FARIAS, M. S.; NEDJAH, N.; MOURELLE, L. d. M. A hardware architecture for subtractive clustering. *International Journal of High Performance Systems Architecture (Print)*, v. 3, p. 167–173, 2011.

- FARIAS, M. S.; NEDJAH, N.; MOURELLE, L. d. M. Radionuclide identification using subtractive clustering. In: *Proceedings of the 2011 International Nuclear Atlantic Conference*. Belo Horizonte: Associação Brasileira de Energia Nuclear - ABEN Press, 2011. p. 387–398.
- FARIAS, M. S.; NEDJAH, N.; MOURELLE, L. d. M. Reconfigurable hardware to radionuclide identification using subtractive clustering. In: *Algorithms and Architectures for Parallel Processing*. Melbourne: In: International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP), 2011. p. 387–398.
- FILEV, D.; ANGELOV, P. Algorithms for real-time clustering and generation of rules from data. In: *Advances in Fuzzy Clustering and its Applications*. England: John Wiley and Sons, Ltd, 2007. p. 355–369.
- GAN, G.; MA, C.; WU, J. *Data Clustering - Theory, Algorithms and Applications*. [S.l.]: Society for Industrial and Applied Mathematics, 2007.
- GRUPEN, C. *Introduction to Radiation Protection*. Siegen, Germany: Springer Berlin Heidelberg, 2010.
- HATHAWAY, J. B. R.; HU, Y. Generalized fuzzy c-means clustering strategies using lp norm distances. In: *Proc. Of SPIE Conf.on Application of Fuzzy Logic Technology*. [S.l.: s.n.], 1993. p. 246–254.
- HENRIKSEN, T.; MAILLIE, H. D. *Radiation and Health*. USA: Taylor and Francis, 2003.
- HINES, J. W.; WREST, D. J.; UHRIG, R. E. Signal validation using an adaptative neural fuzzy inference system. *Nuclear Technology*, 1997.
- JAIN, A. K.; DUBES, R. C. *Algorithms for Clustering Data*. [S.l.]: Prentice Hall, 1988. 227–276 p.
- JAIN, A. K.; MURTY, M. N.; FLYNN, P. *Data Clustering: A Review*. [S.l.]: ACM Computing Surveys, 1999. 264–323 p.
- KAHN, B. *Environmental radiation branch*. Atlanta, GA 30332: Georgia Tech Research Institute, 2006. 162–188 p.
- KAUFMAN, L.; ROUSSEEUW, P. *Finding Groups in Data: An Introduction to Cluster Analysis*. New York, NY: John Wiley and Sons, 1990. 34–42 p.

- KLEINRATH, V. et al. Application of nucleonica's gamma spectrum generator and easy montecarlo simulation tools on nuclear security issues. *Nuclear Science Symposium Conference Record (NSS/MIC), 2010 IEEE*, p. 900 – 901, 2010.
- KNOLL, G. F. *Radiation Detection and Measurement*. New York: John Wiley and Sons, 1989.
- KRUSE, R.; DORING, C.; LESOT, M. *Advances in Fuzzy Clustering and its Applications*. England: John Wiley and Sons, Ltd, 2007. 3-30 p.
- MACQUEEN, J. B. Some methods for classification and analysis of multivariate observations. In: BERKELEY, U. o. C. P. (Ed.). *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*. [S.l.: s.n.], 1967. v. 23, n. 1-3, p. 15-29.
- MATHWORKS. *Fuzzy Logic Toolbox For Use With MATLAB*. [S.l.]: The MathWorks Inc., 1999.
- MENTOR-GRAPHICS. *ModelSim User's Manual*. Oregon, USA: Mentor-Graphics, 2011.
- NAVABI, Z. *VHDL: analysis and modeling of digital systems*. New York, NY, USA: McGraw Hill, 1997. 656 p. (Second edition).
- NING, J. N.; CHOU, H. P. Construction and evaluation of fault detection network for signal validation. *IEEE Transactions on Nuclear Science*, 1992.
- NOUAILHETAS, Y. *Radiações Ionizantes e a vida*. RJ: CNEN, 2003.
- OLIVEIRA, J. V.; PEDRYCZ, W. *Advances in Fuzzy Clustering and its Applications*. England: John Wiley and Sons, Ltd, 2007. 5-30 p.
- OLIVEIRA, M. V. *Metodologia para validação de sinal usando modelos empíricos com técnicas de inteligência artificial aplicada a um reator nuclear*. Dissertação — UFRJ, 2005.
- QUANTRAD. *Ranger Operation Manual*. [S.l.], 1998.
- RIPLEY, B. D. *Pattern recognition and neural networks*. Cambridge: Cambridge University Press, 1996. 2552-2555 p.
- RUGGIERO, M. A. G.; LOPES, V. L. R. *Calculo numérico: aspectos teóricos e computacionais*. Brasil: McGraw Hill, 1988. 295 p.
- RUNKLER, T. A. Relational fuzzy clustering. In: *Advances in Fuzzy Clustering and its Applications*. England: John Wiley and Sons, 2007. p. 32-51.

- SAIC. *RadSmart*. [S.l.], 2000. Manual.
- SAIC. *EXPLORANIUM GR-135 - Identifier*. [S.l.], 2009. Manual.
- SANTI-JONES, P.; GU, D. Fractional fixed point neural networks: an introduction. 2008. (Department of Computer Science, University of Essex, Wivenhoe Park, Colchester, Essex CO4 3SQ).
- SHARAN, R.; SHAMIR, R. Click: A clustering algorithm with applications to gene expression analysis. In: *ISMB-2000 Proceedings*. [S.l.]: American Association for Artificial Intelligence, 2000. p. 307–316.
- SILVA, R. M. *Implementação em hardware de redes neurais artificiais com topologia configurável*. Dissertação — UERJ, 2010.
- SILVA, R. M. da; NEDJAH, N.; MOURELLE, L. M. Reconfigurable mac-based architecture for parallel hardware implementation on fpgas of artificial neural networks using fractional fixed point representation. In: *Procs. of the 19th ICANN09, International Conference on Artificial Neural Networks*. Limassol, Cyprus: Springer Berlin, 2009. v. 19, p. 475–484.
- TANENBAUM, A. S. *Organização estruturada de computadores*. Brasil: Prentice Hall Brasil, 2007.
- TAUHATA, L.; SALATI, I. *Fundamentos de Radioproteção e Dosimetria*. RJ: IRD/CNEN, 2003.
- THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern recognition*. San Diego, CA: Academic Press, 2006.
- TSOULFANIDIS, N. *Measurement and Detection of Radiation*. USA: Taylor and Francis, 1995.
- TURNER, J. E. *Atoms, Radiation, and Radiation Protection*. Germany: Wiley-VCH, 2007.
- WILLETT, P. Recent trends in hierarchical document clustering: A critical review. *Information Processing and Management*, v. 24(5), p. 577–597, 1988.
- WISHART, D. K-means clustering with outlier detection, mixed variables and missing values. In: *Exploratory Data Analysis in Empirical Research*. New York: Schwaiger, M., and Opitz, O., editors, 2002. p. 216–226.



- XILINX. *MicroBlaze Processor Reference Guide*. USA: Xilinx, 2005.
- XILINX. *Using Block RAM in Spartan-3 Generation FPGAs*. [S.l.], 2005.
- XILINX. *Embedded System Tools Reference Manual - EDK*. [S.l.], 2008.
- XILINX. *Spartan 3E FPGA Family*. [S.l.]: Xilinx, 2009.
- XILINX. *Spartan-3E FPGA Starter Kit Board - User Guide*. [S.l.]: Xilinx, 2011.
- XILINX. *ISE 10.1 Quick Start Tutorial*. [S.l.], 2012.
- XILINX. *Virtex-6 Family Overview*. [S.l.], 2012.
- XU, R.; WUNSCH, D. C. *Clustering*. Piscataway, NJ: IEEE, 2009.
- YAGER, R.; FILEV, D. Learning of fuzzy rules by mountain clustering. *Proc. of SPIE Conf on Application of Fuzzy Logic Technology*, Boston, MA, p. 246–254, 1993.
- YEUNG, K.; MEDVEDOVIC, M.; BUMGARNER, R. Clustering gene-expression data with repeated measurements. *Genome Biology*, G34, p. 1–17, 1998.
- ZADEH, L. *Fuzzy sets*. [S.l.]: Information and Control, 1965. 338-353 p.
- ZHAO, Y.; KARYPIS, G. Evaluation of hierarchical clustering algorithms for document datasets. In: *CIKM '02 Proceedings of the eleventh international conference on Information and knowledge management*. [S.l.]: ACM New York, 2002. p. 141–168.